Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

1998-12

Modeling, simulation and visualization of aerocapture

Leszczynski, Zigmond V.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/32660

Downloaded from NPS Archive: Calhoun

http://www.nps.edu/library

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

MODELING, SIMULATION AND VISUALIZATION OF AEROCAPTURE

by

Zigmond V. Leszczynski

December 1998

Thesis Advisor: I. Michael Ross

19990119 027

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 1998	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE
MODELING, SIMULATION AND VISUALIZATION OF AEROCAPTURE

5. FUNDING NUMBERS

6. AUTHOR(S) Zigmond V. Leszczynski

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109-8099

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

A vehicle travelling from Earth to another planet on a ballistic trajectory approaches that planet at hyperbolic velocity. Upon arrival, the vehicle must significantly reduce its speed for orbit insertion. Traditionally, this deceleration has been achieved by propulsive capture, which consumes a large amount of propellant. Aerocapture offers a more fuel-efficient alternative by exploiting vehicular drag in the planet's atmosphere. However, this technique generates extreme heat, necessitating a special thermal protection shield (TPS). Performing a trade study between the propellant mass required for propulsive capture and the TPS mass required for aerocapture can help determine which method is more desirable for a particular mission. The research objective of this thesis was to analyze aerocapture dynamics for the advancement of this trade study process. The result was an aerocapture simulation tool (ACAPS) developed in MATLAB with SIMULINK, emphasizing code validation, upgradeability, user-friendliness and trajectory visualization. The current version, ACAPS 1.1, is a three-degrees-of-freedom point mass simulation model that incorporates a look-up table for the Mars atmosphere. ACAPS is expected to supplement the National Aeronautics and Space Administration (NASA) Jet Propulsion Laboratory (JPL) Project Design Center (PDC) toolkit as preliminary design software for the Mars 2005 Sample Return (MSR) Mission, Mars 2007 Mission, Mars Micromissions, Neptune/Triton Mission, and Human Mars Mission.

14. SUBJECT TERMS Aerocapture Simulation (ACAPS), Aeroassist, National Aeronautics and Space Administration (NASA), Jet Propulsion Laboratory (JPL), Human Mars Mission, Mars 2001 Mission, Mars 2005 Sample Return (MSR) Mission, Mars 2007 Mission, Mars Micromissions, Neptune/Triton Mission, MATLAB, SIMULINK

15. NUMBER OF PAGES **108**

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

Approved for public release; distribution is unlimited.

MODELING, SIMULATION AND VISUALIZATION OF AEROCAPTURE

Zigmond V. Leszczynski
Lieutenant Commander (Select), United States Navy
B.S., United States Naval Academy, 1989

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1998

Author: [redacted]
Zigmond V. Leszczynski

Approved by: [redacted]
I. Michael Ross, Thesis Advisor

[redacted]
Steven E. Matousek, Second Reader

[redacted]
Gerald H. Lindsey, Chairman
Department of Aeronautics and Astronautics

ABSTRACT

A vehicle travelling from Earth to another planet on a ballistic trajectory approaches that planet at hyperbolic velocity. Upon arrival, the vehicle must significantly reduce its speed for orbit insertion. Traditionally, this deceleration has been achieved by propulsive capture, which consumes a large amount of propellant. Aerocapture offers a more fuel-efficient alternative by exploiting vehicular drag in the planet's atmosphere. However, this technique generates extreme heat, necessitating a special thermal protection shield (TPS). Performing a trade study between the propellant mass required for propulsive capture and the TPS mass required for aerocapture can help determine which method is more desirable for a particular mission. The research objective of this thesis was to analyze aerocapture dynamics for the advancement of this trade study process. The result was an aerocapture simulation tool (ACAPS) developed in MATLAB with SIMULINK, emphasizing code validation, upgradeability, user-friendliness and trajectory visualization. The current version, ACAPS 1.1, is a three-degrees-of-freedom point mass simulation model that incorporates a look-up table for the Mars atmosphere. ACAPS is expected to supplement the National Aeronautics and Space Administration (NASA) Jet Propulsion Laboratory (JPL) Project Design Center (PDC) toolkit as preliminary design software for the Mars 2005 Sample Return (MSR) Mission, Mars 2007 Mission, Mars Micromissions, Neptune/Triton Mission, and Human Mars Mission.

DTIC QUALITY INSPECTED 3

DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered completely validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

I. INTRODUCTION 1
A. BACKGROUND 1
1. Aeroassist History 1
2. Aerocapture Overview 2
3. Current Research 3
B. MATLAB AND SIMULINK 4
II. MODELING 5
A. EQUATIONS OF MOTION 5
1. Lift and Drag 7
2. Accelerations 7
3. Rotating Atmosphere 7
4. Canonical Forms 8
B. VEHICLE 12
C. PLANET 13
1. Atmospheric Density 13
2. Gravitational Field 13
D. COORDINATE SYSTEMS 17
E. FIDELITY 17
III. COMPUTER CODE 19
A. MATLAB 19
B. SIMULINK 20
IV. CODE VALIDATION 23
A. EXOATMOSPHERIC 23
B. ENDOATMOSPHERIC 30
C. MARS MISSION 2001 33
V. ANALYSIS OF MARS MISSION 2007 AEROCAPTURE 39
A. FINAL STATE VECTOR 39
1. Visualization 42
2. Stagnation Point Heating 45
3. Accelerations 45
B. ORBITAL ANALYSIS 47
1. Visualization 47
2. Classical Elements 49
3. APOAPSE BURN AND ΔV 49
C. CORRIDOR DEFINITION 49
D. FLIGHT PATH ANGLE DETERMINATION 51
E. REQUIRED FIDELITY 52
VI. SUMMARY AND RECOMMENDATIONS 59
A. ACAPS 59

B. FUTURE RESEARCH....59
VII. ACAPS USER'S MANUAL....61
A. ACAPS OVERVIEW....61
1. Description....61
2. Equations of Motion....61
3. MATLAB....61
4. SIMULINK....62
5. Installation....63
6. Getting Started....63
B. GRAPHIC USER INTERFACE (GUI)....64
1. Flight Plan....64
2. Preflight....66
3. Flight....67
4. Postflight....68
5. Design....70
6. Visualize....71
C. MODEL MODIFICATIONS....74
1. General....74
2. Planets....74
3. Spacecraft....74
4. Guidance....74
5. Atmosphere....75
6. Gravity....75
7. Output....76
D. EXTRAS....76
1. Start/Pause....76
2. Discontinuities....76
APPENDIX A: CORRIDOR FUNCTION OUTPUT....77
APPENDIX B: GAMMA FUNCTION OUTPUT....81
LIST OF REFERENCES....83
BIBLIOGRAPHY....85
INITIAL DISTRIBUTION LIST....89

LIST OF FIGURES

Figure 1: Coordinate System 5
Figure 2: ACAPS Block Diagram 19
Figure 3: SIMULINK Block Diagram 20
Figure 4: Constant Energy 24
Figure 5: Δ Energy 25
Figure 6: Constant Angular Momentum 26
Figure 7: Δ Angular Momentum 27
Figure 8: Constant Eccentricity 28
Figure 9: Δ Eccentricity 29
Figure 10: Ballistic Entry 31
Figure 11: Lifting Entry 32
Figure 12: Mars 2001 Final State Vector 35
Figure 13: Mars 2001 Final State Vector 36
Figure 14: Mars 2001 Acceleration Magnitude 37
Figure 15: Mars 2001 Acceleration Elements 38
Figure 16: Mars 2001 Orbiter Aerocapture Simulation 38
Figure 17: NASA Mars 2001 Orbiter in Aerocapture Shell 39
Figure 18: Mars 2007 Final State Vector 41
Figure 19: Mars 2007 Final State Vector 42
Figure 20: Mars 2007 Aerocapture Profile 43
Figure 21: Mars 2007 Aerocapture Profile Planet Graphic 44
Figure 22: Mars 2007 Stagnation Point Heating 45
Figure 23: Mars 2007 Acceleration Magnitude 46
Figure 24: Mars 2007 Acceleration Elements 47
Figure 25: Mars 2007 Aerocapture Visualization 48
Figure 26: Mars 2007 Aerocapture Corridor 51
Figure 27: Degrees 1 and 5 Fidelity Comparison 54
Figure 28: Degrees 1 and 5 Fidelity Comparison 55
Figure 29: Degrees 2 and 5 Fidelity Comparison 56
Figure 30: Degrees 2 and 5 Fidelity Comparison 57

LIST OF TABLES

Table 1: Conics Parameters 29
Table 2: Fidelity Degrees 66

LIST OF SYMBOLS

A	vehicle reference area
C_D	vehicle coefficient of drag
C_L	vehicle coefficient of lift
g	acceleration due to gravity
h_o	reference altitude
H_p	atmospheric scale height
i	orbital inclination
I_{sp}	specific impulse in seconds
L/D	vehicle lift to drag ratio
m	vehicle mass
M	vehicle initial mass
r	radial distance from planet center
R	planet radius
T	vehicle thrust
v	magnitude of vehicle velocity with respect to the planet
v_e	rocket exhaust speed
α	vehicle angle of attack
δ	vehicle bank angle
γ	vehicle flight path angle
φ	vehicle latitude
μ	planet gravitational constant
θ	vehicle longitude
ρ	atmospheric density
ρ_o	reference atmospheric density

Ω magnitude of the planet's inertial angular velocity

ψ vehicle heading angle

ACKNOWLEDGEMENT

The author would like to thank the following people for their invaluable guidance and assistance provided in the completion of this thesis:

Professor I. Michael Ross, Naval Postgraduate School

Mr. Steven Matousek, NASA Jet Propulsion Laboratory

Mr. David Spencer, NASA Jet Propulsion Laboratory

Mr. Daniel Lyons, NASA Jet Propulsion Laboratory

Ms. Stacy Weinstein, NASA Jet Propulsion Laboratory

Mr. Paul Wercinski, NASA Ames Research Center

Mr. Gary Allen, NASA Ames Research Center

Mr. Bobby Braun, NASA Langley Research Center

I. INTRODUCTION

A. BACKGROUND

The research objective of this thesis was to create a user-friendly tool to analyze aerocapture dynamics for the National Aeronautics and Space Administration's (NASA) Jet Propulsion Laboratory (JPL) managed by the California Institute of Technology. The result was an aerocapture simulation tool called ACAPS (AeroCAPture Simulation). ACAPS is ultimately expected to supplement JPL's Project Design Center (PDC) for Integrated Concurrent Engineering (ICE).

The Naval Postgraduate School (NPS) and JPL are currently incorporating ACAPS into the Mars 2005 Sample Return (MSR) Mission design with France's Centre National d'Etudes Spatiales (CNES). It is anticipated that ACAPS will also serve as preliminary design software for JPL's Mars 2007 Design Team, and possibly for the Mars Micromissions and Human Mars Mission. Consequently, emphasis was placed on code validation, upgradeability, user-friendliness and trajectory visualization.

1. Aeroassist History

Aeroassist has been part of space exploration since the first spacecraft rocketed into orbit. Aeroassist leverages a planet's atmosphere to accomplish propellant-saving maneuvers, which improve or enable a mission. Aeroassist includes aerobraking, direct entry (re-entry), aerogravity assist, synergetic maneuvering, and aerocapture. Aerobraking uses a planet's outer atmosphere over a period of months to reduce the semimajor axis by generating drag on the spacecraft. It is usually applied after a propulsive capture to lower the apoapse. Direct entry decelerates a spacecraft enroute to touchdown with a planet's atmosphere over a short period of time, like the space shuttle. Aerogravity assist is similar to gravity assist, except that the spacecraft descends to lower

altitudes, yielding a greater momentum change [Ref. 6]. Spacecraft could apply this at planet fly-by, during interplanetary transit.

Direct entry, or re-entry, was performed on Mercury and Gemini, followed by Apollo, Viking, Pioneer-Venus and Galileo. Direct entry is also planned for future missions. The Mars Microprobe, DS-2, will directly enter the atmosphere after release from the Mars 98 lander a few tens of minutes from atmospheric interface; the Stardust Sample Return Capsule will enter the Earth's atmosphere with a direct entry; and the Huygens probe will directly enter the Titan atmosphere. Magellan was the first spacecraft to perform aerobraking when it circularized its orbit about Venus. Mars Global Surveyor (MGS) is currently aerobraking to circularize its orbit. Aerogravity assist has never been performed, nor are there any currently planned missions. Aerocapture has also never been performed. A technology demonstration was planned for the Mars 2001 Orbiter, but recent changes to the mission architecture due to budget constraints called for cancellation of that mission objective. The first aerocapture may occur with the Mars 2005 Orbiter. Currently planned missions that might also require aerocapture to make them feasible include the Mars 2007 Mission, the Mars Micromissions, the Neptune/Triton Mission, and the Human Mars Mission. [Ref. 6]

2. Aerocapture Overview

Aerocapture employs the planet's atmosphere to decelerate a vehicle from the hyperbolic approach of interplanetary transit to orbital velocity about the planet. The other option is propulsive capture, which has the greatest propellant demand over any other maneuver during most interplanetary missions. Thus, aerocapture conserves a great amount of propellant for the orbit insertion procedure. "Key variables [affecting aerocapture] include atmospheric structure and composition, desired orbit geometry, interplanetary approach accuracy, entry velocity, and vehicle L/D." [Ref. 6]

Extreme heats generated during aerocapture require special thermal protection shields (TPS) for energy dissipation. A trade study between the TPS mass and the mass of propellant for propulsive capture can help determine which method is more desirable for any given mission. Another design consideration for aerocapture is the autonomous guidance, navigation and control system, which must respond robustly to unexpected atmospheric density changes.

Aerocapture may be a necessity for future space travel. The Human Mars Mission, for example, may require aerocapture. The spacecraft will support a human crew for many months in transit and consequently will be very massive. The amount of propellant required to decelerate by propulsive capture may be weight-prohibitive at launch. Employing aerocapture might save enough propellant mass at launch to make the mission possible.

3. Current Research

Current aerocapture related research involves, among other things, control algorithm development, atmospheric modeling, and thermal protection shield design. Jet Propulsion Laboratory, Lockheed Martin Space Systems, and NASA Langley are foremost in aerocapture control algorithm development. An adaptable control algorithm is required to accurately guide the spacecraft through aerocapture, despite unexpected atmospheric perturbations. Inability to respond quickly with correct control inputs could cause the spacecraft to burn up in the atmosphere or bounce off into space. Langley tests control algorithms with six-degrees-of-freedom (6-DOF) Monte Carlo simulations, in which the atmospheric density is modeled with MarsGRAM. NASA Marshall Research Center developed, and is continually improving, MarsGRAM with new Mars Global Surveyor data and ground-based Earth observations. NASA Ames is researching the next generation thermal protection shields. Light and heat resilient thermal protection shields will make many future missions practicable. Ames and Langley are coordinating a web-

based environment called the Integrated Design Systems (IDS). IDS will combine the most advanced control algorithms, atmospheric modeling and thermal protection shield composites into a virtual spacecraft design environment for re-entry and aerocapture missions [Ref. 1].

Much work is being done in aerocapture, some of which is mentioned above. However, it appears that there is no fast, first-cut, aerocapture simulation tool for preliminary mission design trade studies. ACAPS will save time and money in the mission design phase by filling this gap, thereby upholding the NASA motto for future missions: "Faster, Better, Cheaper."

B. MATLAB AND SIMULINK

MATLAB 5.x was selected for the development of the computer simulation tool for several reasons. First, it allows fast graphical user interface (GUI) construction with a new graphic user interface layout tool that was recently introduced in MATLAB 5.x. Graphic user interfaces greatly enhance user-friendliness, which is one of the developmental objectives. Second, MATLAB has a simulation environment toolbox called SIMULINK, which contains state-of-the-art numerical integrators ideal for solving the non-linear, forward propagating, ordinary differential equations of motion (EOM) that are at the heart of ACAPS. Third, MATLAB code is platform independent, which also enhances user-friendliness. Fourth, MATLAB is generally easier to understand, troubleshoot and modify than C or FORTRAN. Also, it takes fewer lines of MATLAB to accomplish a given task than it would in C, for example. According to Risk International, approximately 875 lines of MATLAB code equates to 5000 lines of C code [Ref. 5]. ACAPS contains roughly 4200 lines of MATLAB code, or an equivalent of 24,000 lines of C code.

II. MODELING

Modeling, as used here, is the mathematical description of the space system in the form of equations of motion to represent the vehicle in a dynamic environment. ACAPS modeling encompasses the vehicular lift and drag, atmospheric density, and gravitational field about the planet.

A. EQUATIONS OF MOTION

Equations of motion describe the location and velocity of the spacecraft during aerocapture. The coordinate system in which the equations of motion are defined is given in Figure 1 [Ref. 7].

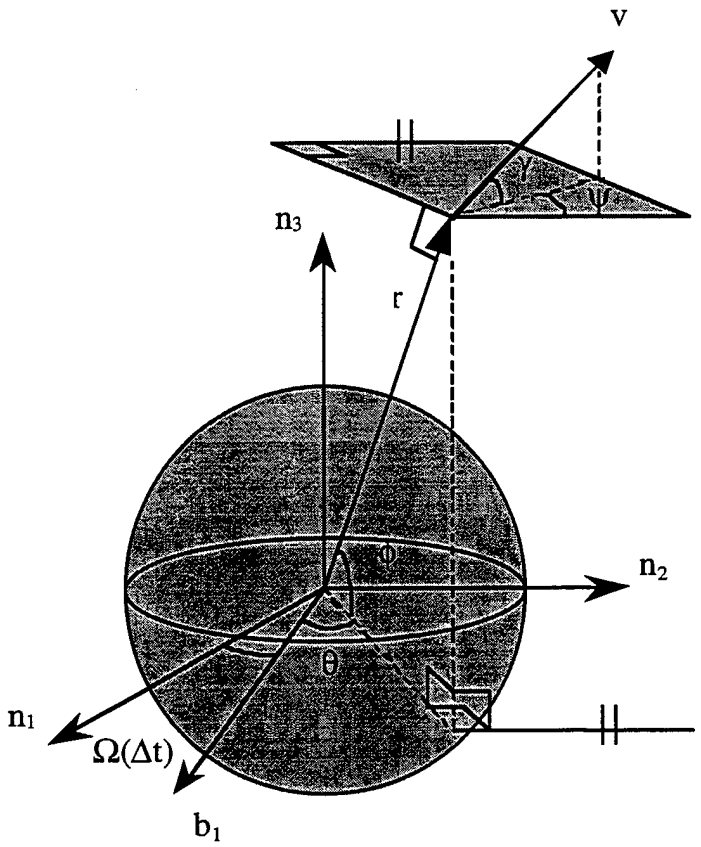

Figure 1: Coordinate System

The initial state vector is, $\mathbf{x} = [r, \theta, \varphi, v, \psi, \gamma, m]^T$, where [Ref. 7],

r ≡ radial position

θ ≡ longitude (positive eastward, along the equatorial plane)

φ ≡ latitude (positive northward, along the meridian)

v ≡ magnitude of the velocity of the spacecraft with respect to the planet

ψ ≡ heading angle

γ ≡ flight path angle

m ≡ mass (included for later implementation of body-fixed thrust)

The equations of motion for ACAPS were developed systematically, progressing from simple to complex. The non-rotating atmosphere was first described, programmed, and validated. The equations of motion for a rotating atmosphere were then programmed into ACAPS, which included centrifugal and Coriolis forces. The equations of motion for a non-rotating atmosphere with an inverse-square gravity field are defined as follows, [Ref. 10]:

$$\dot{r} = v \sin\gamma \tag{1}$$

$$\dot{\theta} = \frac{v \cos\gamma \cos\psi}{r \cos\varphi} \tag{2}$$

$$\dot{\varphi} = \frac{v \cos\gamma \sin\psi}{r} \tag{3}$$

$$\dot{v} = a_s - g \sin\gamma \tag{4}$$

$$\dot{\psi} = \frac{a_w}{v \cos\gamma} - \frac{v}{r} \cos\gamma \cos\psi \tan\varphi \tag{5}$$

$$\dot{\gamma} = \frac{a_n}{v} - \frac{1}{v}\left(g - \frac{v^2}{r}\right)\cos\gamma \tag{6}$$

$$\dot{m} = -\frac{T}{v_e} \tag{7}$$

1. Lift and Drag

The aerodynamic lift and drag are given by [Ref. 7],

$$L = 0.5\rho(r,\theta,\varphi,t)v^2 AC_L(\alpha) \tag{8}$$

$$D = 0.5\rho(r,\theta,\varphi,t)v^2 AC_D(\alpha) \tag{9}$$

where ρ, A, C_L, C_D, and α are respectively the atmospheric density, the vehicle reference area, lift coefficient, drag coefficient, and angle of attack.

2. Accelerations

G-loading is described by a_s a_n a_w, the non-gravitational perturbing accelerations on the vehicle in the tangential, normal, and binormal directions respectively, expressed in the Frenet frame [Ref. 7],

$$a_s = \frac{T\cos\alpha - D}{m} \tag{10}$$

$$a_n = \frac{(L + T\sin\alpha)\cos\delta}{m} \tag{11}$$

$$a_w = \frac{(L + T\sin\alpha)\sin\delta}{m} \tag{12}$$

where T and δ are respectively the body-fixed thrust, and bank angle.

3. Rotating Atmosphere

Several assumptions constrain the derivation of the seven equations of motion when considering a rotating atmosphere [Ref. 7]:

1. The planet is rotating about the n_3-axis with constant angular speed, Ω (rad/s).
2. The atmosphere is rotating with the planet.
3. The gravity field is given by a scalar potential, $U = U(r,\theta,\varphi)$.

The equations of motion for a rotating atmosphere are defined in terms of Figure 1 as follows [Ref. 7]:

$$\dot{r} = v\sin\gamma \tag{13}$$

$$\dot{\theta} = \frac{v\cos\gamma\cos\psi}{r\cos\varphi} \tag{14}$$

$$\dot{\varphi} = \frac{v\cos\gamma\sin\psi}{r} \tag{15}$$

$$\dot{v} = a_s + g_s + cf_V \tag{16}$$

$$\dot{\psi} = \frac{a_w + g_w}{v\cos\gamma} - \frac{v}{r}\cos\gamma\cos\psi\tan\varphi + cf_\psi + co_\psi \tag{17}$$

$$\dot{\gamma} = \frac{a_n + g_n}{v} + \frac{v\cos\gamma}{r} + cf_\gamma + co_\gamma \tag{18}$$

$$\dot{m} = -\frac{T}{v_e} \tag{19}$$

The centrifugal and Coriolis force terms result from the rotating atmosphere and are given by [Ref. 7]:

$$cf_V = \Omega^2 r\cos\varphi(\sin\gamma\cos\varphi - \cos\gamma\sin\varphi\sin\psi) \tag{20}$$

$$cf_\psi = -\frac{\Omega^2 r}{v\cos\gamma}\sin\varphi\cos\varphi\cos\psi \tag{21}$$

$$co_\psi = 2\Omega(\tan\gamma\cos\varphi\sin\psi - \sin\varphi) \tag{22}$$

$$cf_\gamma = \frac{\Omega^2 r}{v}\cos\varphi(\cos\gamma\cos\varphi + \sin\gamma\sin\varphi\sin\psi) \tag{23}$$

$$co_\gamma = 2\Omega\cos\varphi\cos\psi \tag{24}$$

4. Canonical Forms

The seven equations of motion contain variables that differ in orders of magnitude, e.g. the Mars gravitational constant, $\mu_{Mars} = 3.986x10^{14}\,m^3/s^2$, and the Mars surface atmospheric density, $\rho_s = 3.42x10^{-5}\,kg/m^3$. Numerically combining matrices with such

dissimilar orders of magnitude introduces various errors into the solution of the equations of motion. This is known as an ill-conditioned system. If too severe, an ill-conditioned system could crash the numerical solver.

Implementing canonical equations of motion streamlines the numerical solution process. All variables are non-dimensionalized towards unity to eliminate round-off error and prevent an ill-conditioned system. ACAPS is designed so that the user enters and receives data in the International System of Units (SI), meters, kilograms, and seconds (MKS). Transformations are applied in between the input and output, so the non-linear ordinary differential equation solver computes with all canonical values. This non-dimensionalization also significantly speeds up processing time.

The seven canonical equations of motion are derived by first non-dimensionalizing individual variables as follows,

$$\bar{r} = \frac{r}{R} \tag{25}$$

where R is a specific radius, e.g. the planet radius.

$$\bar{v} = \frac{v}{V} \tag{26}$$

where $V = \sqrt{\frac{\mu}{R}}$

$$\bar{t} = \frac{t}{\tau} \tag{27}$$

where $\tau = \frac{R}{V}$

$$\bar{m} = \frac{m}{M} \tag{28}$$

where M is the spacecraft initial mass. Given these definitions Eqs. (13) through (15) transform as,

$$\dot{\bar{r}} = \frac{d\left(r/R\right)}{d\left(t/\tau\right)} = \frac{\tau}{R}\frac{dr}{dt} = \frac{\tau}{R}[v\sin(\gamma)] = \frac{\tau}{R}[\bar{v}V\sin(\gamma)] = \bar{v}\sin(\gamma) \tag{29}$$

$$\dot{\bar{\theta}} = \frac{d\theta}{d\left(t/\tau\right)} = \tau\frac{d\theta}{dt} = \tau\frac{v}{r}\frac{\cos(\gamma)\cos(\psi)}{\cos(\varphi)} = \frac{\bar{v}}{\bar{r}}\frac{\cos(\gamma)\cos(\psi)}{\cos(\varphi)} \tag{30}$$

$$\dot{\bar{\gamma}} = \frac{d\gamma}{d\left(t/\tau\right)} = \tau\frac{d\varphi}{dt} = \tau\frac{v}{r}\cos(\gamma)\sin(\psi) = \frac{\bar{v}}{\bar{r}}\cos(\gamma)\sin(\psi) \tag{31}$$

The nondimensional effective spacecraft area, related to C_L and C_D, is defined by,

$$\bar{A} = \frac{A}{\left(M/\rho_o R\right)} \tag{32}$$

where ρ_o is the reference atmospheric density, e.g. surface atmospheric density or the atmospheric density at a specific scale height.

The nondimensional lift and drag derived from Eqs. (8) through (9) are given by,

$$\bar{L} = 0.5\bar{\rho}\bar{v}^2\bar{A}C_L(\alpha) \tag{33}$$

$$\bar{D} = 0.5\bar{\rho}\bar{v}^2\bar{A}C_D(\alpha) \tag{34}$$

where the nondimensional atmospheric density is defined by,

$$\bar{\rho} = \frac{\rho}{\rho_o} \tag{35}$$

The nondimensional non-gravitational perturbing accelerations derived from Eqs. (10) through (12) are given by,

$$\bar{a}_s = \frac{\bar{T}\cos\alpha - \bar{D}}{\bar{m}} \tag{36}$$

$$\bar{a}_n = \frac{\left(\bar{L} + \bar{T}\sin\alpha\right)\cos\delta}{\bar{m}} \tag{37}$$

$$\bar{a}_w = \frac{\left(\bar{L} + \bar{T}\sin\alpha\right)\sin\delta}{\bar{m}} \tag{38}$$

where the nondimensional body-fixed thrust is defined by,

$$\bar{T} = \frac{T}{Mg_o} \tag{39}$$

and the gravitational force is defined by,

$$g_o = \frac{\mu}{R^2} \tag{40}$$

The nondimensional planet rotational velocity about its axis is defined by,

$$\overline{\Omega} = \Omega\tau \tag{41}$$

The nondimensional centrifugal and Coriolis force perturbations derived from Eqs. (20) through (24) are given by,

$$\overline{cf_V} = \overline{\Omega^2}\,\bar{r}\cos\varphi(\sin\gamma\cos\varphi - \cos\gamma\sin\varphi\sin\psi) \tag{42}$$

$$\overline{cf_\psi} = -\frac{\overline{\Omega^2}\,\bar{r}}{\bar{v}\cos\gamma}\sin\varphi\cos\varphi\cos\psi \tag{43}$$

$$\overline{co_\psi} = 2\overline{\Omega}(\tan\gamma\cos\varphi\sin\psi - \sin\varphi) \tag{44}$$

$$\overline{cf_\gamma} = \frac{\overline{\Omega^2}\,\bar{r}}{\bar{v}}\cos\varphi(\cos\gamma\cos\varphi + \sin\gamma\sin\varphi\sin\psi) \tag{45}$$

$$\overline{co_\gamma} = 2\overline{\Omega}\cos\varphi\cos\psi \tag{46}$$

The nondimensional exhaust velocity is defined by,

$$\overline{v_e} = \frac{v_e}{V} \tag{47}$$

Eqs. (16) through (18) convert to canonical form by substituting the non-dimensionalized equations and terms above,

$$\dot{\bar{v}} = \overline{a_s} + \overline{g_s} + \overline{cf_V} \tag{48}$$

$$\dot{\bar{\psi}} = \frac{\overline{a_w} + \overline{g_w}}{\bar{v}\cos\gamma} - \frac{\bar{v}}{\bar{r}}\cos\gamma\cos\psi\tan\varphi + \overline{cf_\psi} + \overline{co_\psi} \tag{49}$$

$$\dot{\bar{\gamma}} = \frac{\overline{a_n} + \overline{g_n}}{\bar{v}} + \frac{\bar{v}\cos\gamma}{\bar{r}} + \overline{cf_\gamma} + \overline{co_\gamma} \tag{50}$$

$$\dot{\bar{m}} = -\frac{\overline{T}}{v_e} \tag{51}$$

The non-rotating atmosphere equations of motion, Eqs. (1) through (7), similarly convert to the following canonical forms,

$$\dot{\bar{r}} = \bar{v}\sin\gamma \tag{52}$$

$$\dot{\bar{\theta}} = \frac{\bar{v}\cos\gamma\cos\psi}{\bar{r}\cos\varphi} \tag{53}$$

$$\dot{\varphi} = \frac{\bar{v}\cos\gamma\sin\psi}{\bar{r}} \tag{54}$$

$$\dot{\bar{v}} = \bar{a}_s - \bar{g}\sin\gamma \tag{55}$$

$$\dot{\psi} = \frac{\bar{a}_w}{\bar{v}\cos\gamma} - \frac{\bar{v}}{\bar{r}}\cos\gamma\cos\psi\tan\varphi \tag{56}$$

$$\dot{\gamma} = \frac{\bar{a}_n}{\bar{v}} - \frac{1}{\bar{v}}\left(\bar{g} - \frac{\bar{v}^2}{\bar{r}}\right)\cos\gamma \tag{57}$$

$$\dot{\bar{m}} = -\frac{\bar{T}}{v_e} \tag{58}$$

B. VEHICLE

Vehicle stagnation point heating is calculated from the relationship,

$$Q_{stag} = \frac{C}{\sqrt{R_n}}\rho^N v^M \tag{59}$$

where C, N and M are the coefficients for stagnation point heating characteristic to the spacecraft and the planet's atmosphere, and R_n is the spacecraft nose radius. The atmospheric density, ρ, and the velocity, v, are collected during the simulation.

G-loading is calculated from a_s, a_n, and a_w as the numerical solution propagates forward. The g-load magnitude is solved from the tangential, normal and binormal components,

$$|g-load| = \sqrt{a_s^2 + a_n^2 + a_w^2} \tag{60}$$

C. PLANET

1. Atmospheric Density

The atmospheric density, ρ, is either a global exponential, or a look-up table approximation. Both databases come from the MarsGRAM 3.7 Mars atmosphere model and are referenced in 10 km altitude incremental shells. Exponential interpolation provides the density between shells. The exponential density is expressed by,

$$\rho = \rho_o e^{-\left(\frac{(h-h_o)}{H_p}\right)} \tag{61}$$

where ρ_o, h_o and H_p are the reference atmospheric density, altitude and scale height, respectively.

2. Gravitational Field

The gravitational accelerations g_s, g_n, and g_w are determined from the geopotential, $U = U(r, \theta, \varphi)$. The components of the gravity vector along the "Radial, East and North" (REN-frame) directions are given by [Ref. 7],

$$g_r = \frac{\partial U}{\partial r} \tag{62}$$

$$g_\theta = \frac{1}{r\cos\varphi}\frac{\partial U}{\partial \theta} \tag{63}$$

$$g_\varphi = \frac{1}{r}\frac{\partial U}{\partial \varphi} \tag{64}$$

The transformation from the REN-frame to the Frenet frame is given by [Ref. 7],

$$\begin{bmatrix} g_n \\ g_s \\ g_w \end{bmatrix} = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} g_r \\ g_\theta \\ g_\varphi \end{bmatrix} \tag{65}$$

which simplifies to,

$$\begin{bmatrix} g_n \\ g_s \\ g_w \end{bmatrix} = \begin{bmatrix} \cos\gamma & -\sin\gamma\cos\psi & -\sin\gamma\sin\psi \\ \sin\gamma & \cos\gamma\cos\psi & \cos\gamma\sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} g_r \\ g_\theta \\ g_\varphi \end{bmatrix} \tag{66}$$

The expression for the gravitational potential inverse-square law is developed from the derivative of the geopotential,

$$U(r,\theta,\varphi) = \frac{\mu}{r} \tag{67}$$

$$g_r = \frac{\partial U}{\partial r} = -\frac{\mu}{r^2} \equiv -g \tag{68}$$

$$g_\theta = 0 \tag{69}$$

$$g_\varphi = 0 \tag{70}$$

Hence,

$$g_n = -g\cos\gamma \tag{71}$$

$$g_s = -g\sin\gamma \tag{72}$$

$$g_w = 0 \tag{73}$$

Eqs. (13) through (15) are unaffected, while Eqs. (16) through (18) are given by,

$$\dot{v} = a_s - g\sin\gamma + cf_V \tag{74}$$

$$\dot{\psi} = \frac{a_w}{v\cos\gamma} - \frac{v}{r}\cos\gamma\cos\psi\tan\varphi + cf_\psi + co_\psi \tag{75}$$

$$\dot{\gamma} = \frac{a_n}{v} - \left(g - \frac{v^2}{r}\right)\frac{\cos\gamma}{v} + cf_\gamma + co_\gamma \tag{76}$$

The Zonal Harmonic J_2 gravitational model is more sophisticated than the inverse-square law developed above. It accounts for Mars' gravitational departure from a perfect sphere, reflecting the planet's oblate shape [Ref. 9]. The J_2 expression is developed from the definition of the aspherical gravitational potential [Ref. 9],

$$\begin{aligned} U(r,\theta,\varphi) &= \frac{\mu}{r}\sum_{\ell=0}^{\infty} P_\ell[\sin(\varphi)]\left(\frac{R_\oplus}{r}\right)^\ell C_{\ell 0} \\ &+ \frac{\mu}{r}\sum_{\ell=1}^{\infty}\sum_{m=1}^{\ell} P_\ell[\sin(\varphi)]\left(\frac{R_\oplus}{r}\right)^\ell \{C_{\ell m}\cos(m\lambda) + S_{\ell m}\sin(m\lambda)\} \end{aligned} \tag{77}$$

where μ is the planet's gravitational constant, $P_{l,m}[\sin(\phi)]$ is the Legendre function of degree ℓ and order m, C_{lm} and S_{lm} are the gravitational coefficients. This reduces to one of the several common notations for the aspherical potential, adopted by the International Astronomical Union in 1961 [Ref. 9],

$$U(r,\theta,\varphi)=\frac{\mu}{r}[1-\sum_{\ell=2}^{\infty}J_\ell\left(\frac{R_\oplus}{r}\right)^\ell P_\ell[\sin(\varphi)] + \sum_{\ell=2}^{\infty}\sum_{m=1}^{\ell}\left(\frac{R_\oplus}{r}\right)^\ell P_\ell[\sin(\varphi)]\{C_{\ell m}\cos(m\lambda)+S_{\ell m}\sin(m\lambda)\}] \tag{78}$$

Including only the zonal harmonics of the geopotential, Eq. (78) reduces to,

$$U(r,\theta,\varphi)=\frac{\mu}{r}\left[1-\sum_{\ell=2}^{\infty}\left(\frac{R_\oplus}{r}\right)^\ell J_\ell P_\ell[\sin(\varphi)]\right] \tag{79}$$

Let $\ell=2$,

$$U(r,\theta,\varphi)=\frac{\mu}{r}\left[1-\left(\frac{R_\oplus}{r}\right)^2 J_2 P_2[\sin(\varphi)]\right] \tag{80}$$

The Legendre function equals,

$$P_2[\sin(\varphi)]=\frac{1}{2}[3\sin^2(\varphi)-1] \tag{81}$$

Substitute Eq. (81) into Eq. (80), and let $\varepsilon=\frac{J_2 R_\oplus{}^2}{2}$,

$$U(r,\theta,\varphi)=\frac{\mu}{r}\left[1-\frac{\varepsilon}{r^2}(3\sin^2\varphi-1)\right] \tag{82}$$

The gradient in spherical coordinates is given by,

$$\nabla=\bar{i}_r\frac{\partial}{\partial r}+\bar{i}_\theta\frac{1}{r\cos(\varphi)}\frac{\partial}{\partial\theta}+\bar{i}_\varphi\frac{1}{r}\frac{\partial}{\partial\varphi} \tag{83}$$

The gradient of the potential in spherical coordinates is equal to the J_2 gravity terms,

$$g_r=-\frac{\mu}{r^2}+\frac{3\varepsilon\mu}{r^4}[3\sin^2(\varphi)-1] \tag{84}$$

$$g_\theta=0 \tag{85}$$

$$g_\varphi=-\frac{6\varepsilon\mu}{r^4}\sin(\varphi)\cos(\varphi) \tag{86}$$

Similarly, the J_3 gravity term is derived by first setting $\ell=2,3$ in Eq. (78),

$$U(r,\theta,\varphi)=\frac{\mu}{r}\left[1-\left(\frac{R_\oplus}{r}\right)^2 J_2 P_2[\sin(\varphi)]-\left(\frac{R_\oplus}{r}\right)^3 J_3 P_3[\sin(\varphi)]\right] \tag{87}$$

The Legendre function equals,

$$P_3[\sin(\varphi)]=\frac{1}{2}[5\sin^3(\varphi)-3\sin(\varphi)] \tag{88}$$

Let $\varepsilon=\frac{J_2 R_\oplus{}^2}{2}$ and $\varepsilon_2=\frac{J_3 R_\oplus{}^3}{2}$. Substituting Eq. (88) into Eq. (87), and then taking the gradient of the geopotential gives the J_3 gravity terms,

$$g_r=-\frac{\mu}{r^2}+\frac{3\varepsilon\mu}{r^4}[3\sin^2(\varphi)-1]+\frac{4\varepsilon_2\mu}{r^5}[5\sin^3(\varphi)-3\sin(\varphi)] \tag{89}$$

$$g_\theta=0 \tag{90}$$

$$g_\varphi=-\frac{6\varepsilon\mu}{r^4}\sin(\varphi)\cos(\varphi)+\frac{3\varepsilon\mu}{r^{45}}\cos(\varphi)[1-5\sin^2(\varphi)] \tag{91}$$

The J_3 zonal harmonic for Mars is nearly three orders of magnitude less than J_2. J_2 is consequently the dominant gravitational perturbation resulting from the shape of the planet [Ref. 9].

The canonical J_2 gravity terms derived from Eqs. (89) through (91) are given by,

$$\overline{g_r}=\frac{g_r}{\left(\mu/R^2\right)}=-\frac{1}{\overline{r}^2}+\frac{3}{2}\frac{J_2}{\overline{r}^4}[3\sin^2(\varphi)-1] \tag{92}$$

$$\overline{g_\theta}=\frac{g_\theta}{\left(\mu/R^2\right)}=0 \tag{93}$$

$$\overline{g_\varphi}=\frac{g_\varphi}{\left(\mu/R^2\right)}=-3\frac{J_2}{\overline{r}^4}\sin(\varphi)\cos(\varphi) \tag{94}$$

The canonical J_3 gravity terms derived from Eqs. (89) through (91), are given by,

$$\overline{g_r}=\frac{g_r}{\left(\mu/R^2\right)}=-\frac{1}{\overline{r}^2}+\frac{3}{2}\frac{J_2}{\overline{r}^4}[3\sin^2(\varphi)-1]+2\frac{J_3}{\overline{r}^5}[5\sin^3(\varphi)-3\sin(\varphi)] \tag{95}$$

$$\overline{g_\theta}=\frac{g_\theta}{\left(\mu/R^2\right)}=0 \tag{96}$$

$$\overline{g_\varphi} = \frac{g_\varphi}{\left(\mu/R^2\right)} = -3\frac{J_2}{r^4}\sin(\varphi)\cos(\varphi) + \frac{3}{2}\frac{J_3}{r^5}\cos(\varphi)[1-5\sin^2(\varphi)] \qquad (97)$$

D. COORDINATE SYSTEMS

The initial state vector must be input into ACAPS in the planet-fixed coordinate frame. The final state vector is output in this same frame. JPL designates this coordinate frame IORB flag 42, a rotating frame aligned with the meridian of date and equator on the epoch of the state. The velocity is corrected from the inertial coordinate frame by the rotation rate of the central body, and all state vector elements are input in spherical coordinates.

E. FIDELITY

The gravity model (inverse-square, J_2, or J_3), atmospheric density model (global exponential, or look-up table), and the atmosphere model (non-rotating, or rotating) are applied to the seven equations of motion according to simulation fidelity. The user selects fidelity in the input cueing sequence.

III. COMPUTER CODE

A. MATLAB

The ACAPS code is modular; see the block diagram in Figure 2. Over 150 MATLAB files (M-files), or modules, work together with function calls or from the Command Workspace. Modularity lends two big advantages. First, ACAPS is mission upgradeable. For example, to upgrade a planetary model, only the planet module inside the Modeling Toolbox needs to be opened and modified. Second, the approach to developmental troubleshooting is straightforward: identify the nature of the problem; go to the related Toolbox; eliminate unassociated M-files in that Toolbox; test the remaining modules for faults; correct the defective modules.

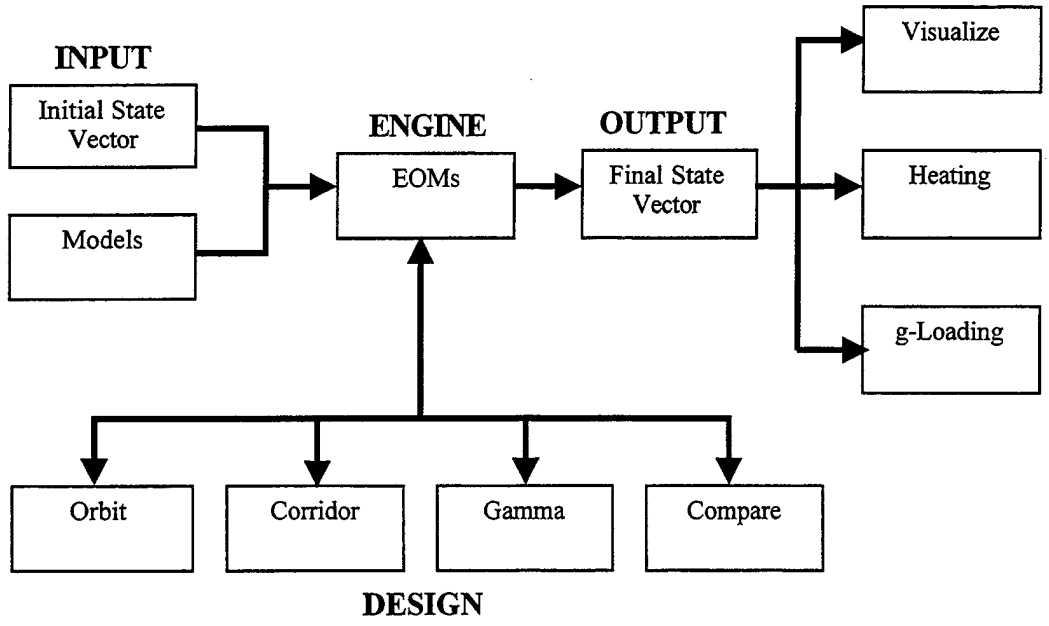

Figure 2: ACAPS Block Diagram

A large code, without modularity, would be burdensome to upgrade or troubleshoot, especially after some time had passed since the developer worked on the

program, or if a second party attempted modifications. Since Jet Propulsion Laboratory design team members or future Naval Postgraduate School thesis students will most likely upgrade ACAPS for future missions, second party modifications are inevitable, making modularity an implicit requirement.

B. SIMULINK

The SIMULINK block diagram in Figure 3 indicates how the equations of motion that describe the spacecraft's trajectory through the atmosphere interact with the whole system. The MATLAB Function block calls the M-file containing the equations of motion. This M-file calls the atmospheric density and gravity subroutines and references model parameters in the Command Workspace to build the equations of motion. The numerical integrator block, $\frac{1}{s}$, then integrates the equations of motion. The initial state vector provides the integrator's initial conditions. The equations of motion propagate forward through the closed loop, reacting to any control inputs from the guidance block. The final state vector is tapped off the loop after the integrator for output.

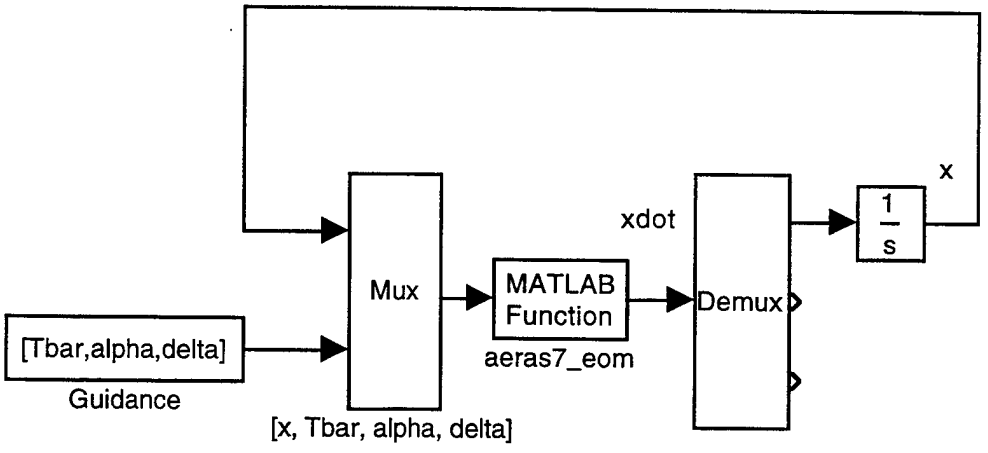

Figure 3: SIMULINK Block Diagram

Seven non-linear, forward propagating, ordinary differential equations (ODE) describe the spacecraft dynamics in three-degrees-of-freedom (3-DOF) during aerocapture. They are solved by numerical integration in SIMULINK. The ordinary differential equation solver works through the simulation by computing a solution to the equations of motion at each time step. If the solution satisfies error tolerance criteria set in SIMULINK, the step size remains the same and the solver continues to the next time step. If the solution does not satisfy the criteria, the solver shrinks the step size and recalculates the solution. The variable-step function modifies step sizes during the simulation, providing error control and zero crossing detection. SIMULINK implements state-of-the-art variable time step control algorithms to optimize performance. The numerical integration algorithm used in ACAPS is Dormand-Prince 45, the highest fidelity ordinary differential equation solver available in SIMULINK. It is an explicit Runge-Kutta (4,5) formula pair. In computing each step, the Dormand-Prince pair only needs the solution from the preceding time step, saving memory and run time [Ref. 8].

IV. CODE VALIDATION

ACAPS code was thoroughly validated and developed progressively through more complex stages to facilitate this validation. First, a two-degrees-of-freedom version was completed and validated; then a three-degrees-of-freedom version was completed and validated. Both of these had non-rotating atmospheres. Next, a two-degrees-of-freedom rotating atmosphere version was constructed and validated. This was soon improved to a rotating atmosphere three-degrees-of-freedom version and validated.

As explained in the following sections, the code was tested outside the atmosphere with several variants; inside the atmosphere with no lift; inside the atmosphere with lift; and against Jet Propulsion Laboratory's predictions of a real world mission. Outside the atmosphere, or exoatmospherically, the mathematical implementation of two-body particle dynamics was verified. Inside the atmosphere, or endoatmospherically, the lift and drag models were verified against two case studies that employed analytic closed form approximations of re-entry vehicle dynamics. Finally, ACAPS was used to simulate the Mars 2001 Orbiter aerocapture, and these results were compared with projections from Jet Propulsion Laboratory. This final test confirmed the modeling of a rotating atmosphere with centrifugal and Coriolis forces in ACAPS.

A. EXOATMOSPHERIC

Four exoatmospheric tests were conducted with ACAPS to verify the following mathematical facts of two-body dynamics without perturbations: constant total specific mechanical energy; constant specific angular momentum; constant eccentricity; and properties of conic sections.

Figure 4 is a plot from an ACAPS simulation with zero atmospheric density that validates a piece of the code. It confirms that the total specific mechanical energy in a conservative two-body system is a constant. Figure 5 plots the difference, $E(0) - E(t)$,

demonstrating the small magnitude of variation that otherwise appears as a constant in Figure 4.

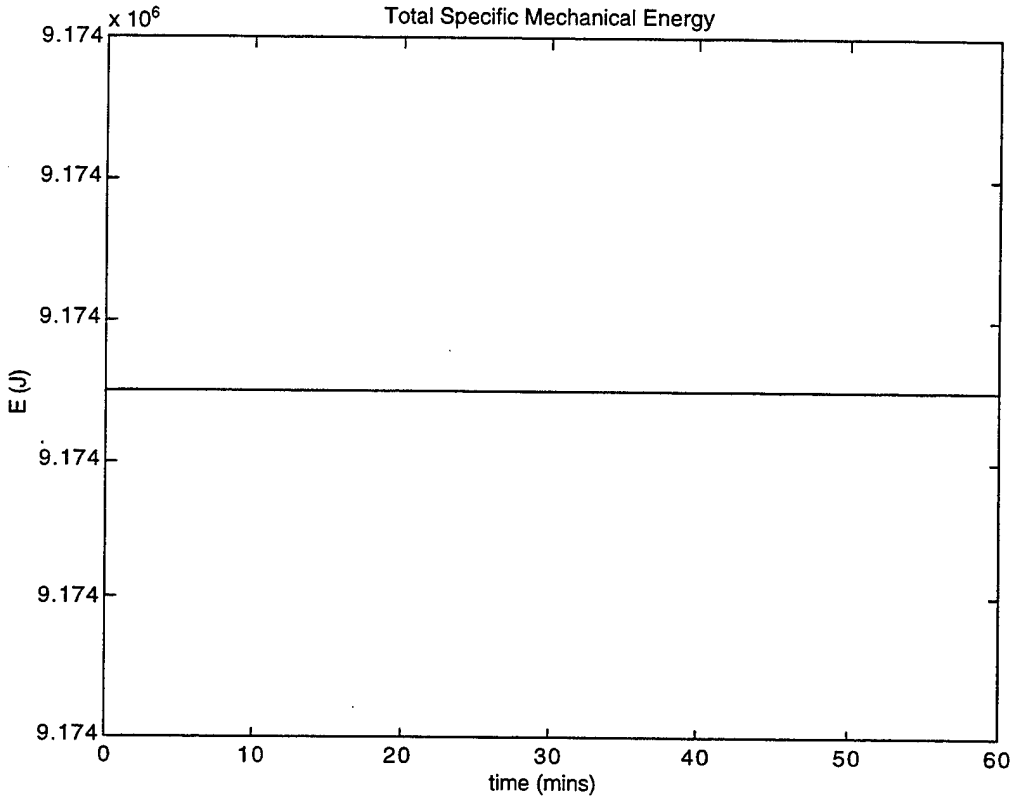

Figure 4: Constant Energy

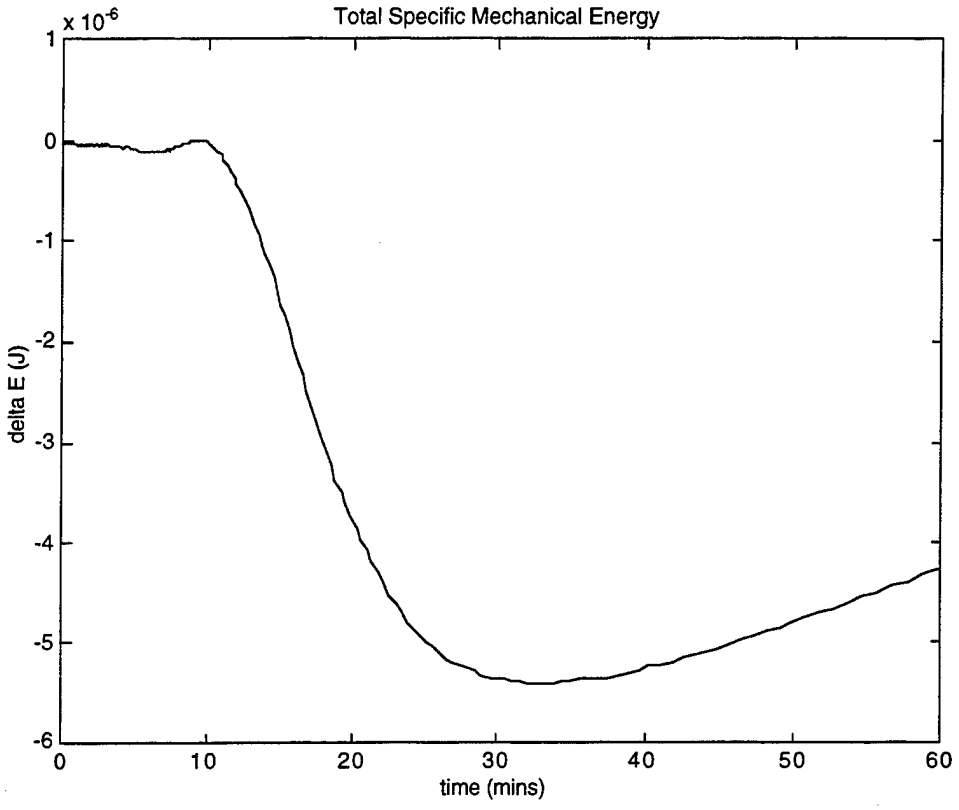

Figure 5: Δ Energy

As shown in Figure 6, an ACAPS simulation with zero atmospheric density also confirmed that it correctly modeled the total specific angular momentum in a conservative two-body system as constant. Figure 7 plots the difference of the first angular momentum value from all the following values, demonstrating the small magnitude of variation that appears constant in Figure 6.

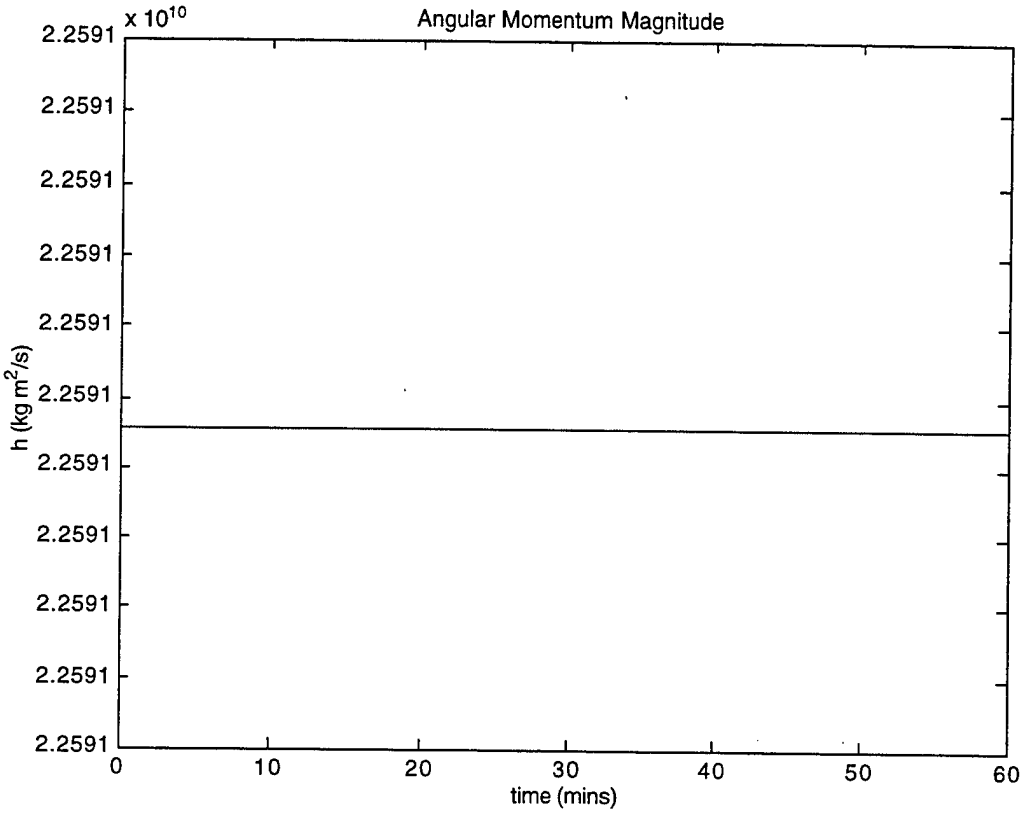

Figure 6: Constant Angular Momentum

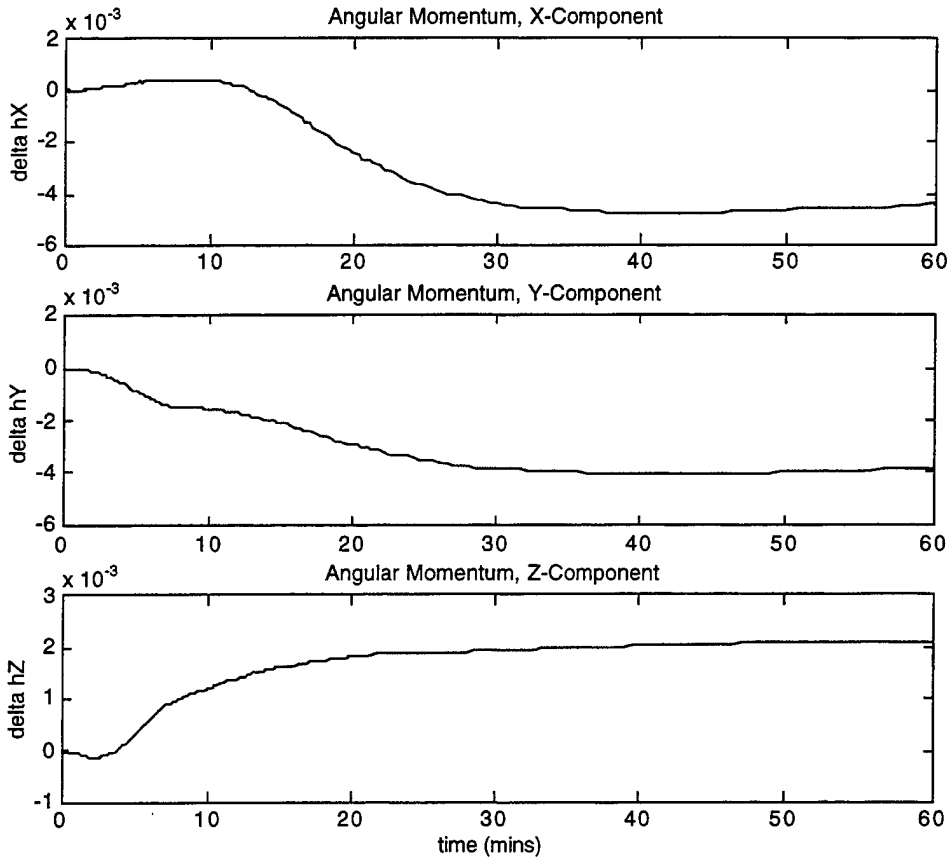

Figure 7: Δ Angular Momentum

An ACAPS simulation with zero atmospheric density confirmed that it correctly modeled the eccentricity of a conic in a conservative two-body system as a constant. Figure 9 plots the difference of the first eccentricity value from all the following values, demonstrating a small magnitude of variation that otherwise appears constant in Figure 8.

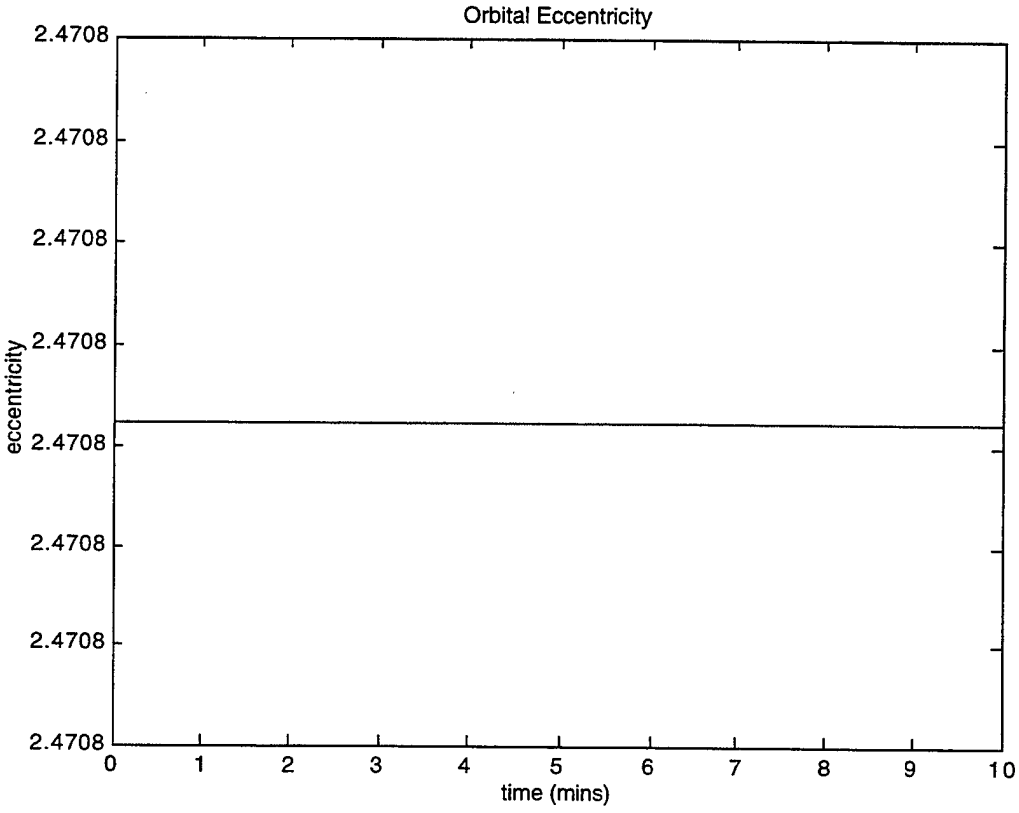

Figure 8: Constant Eccentricity

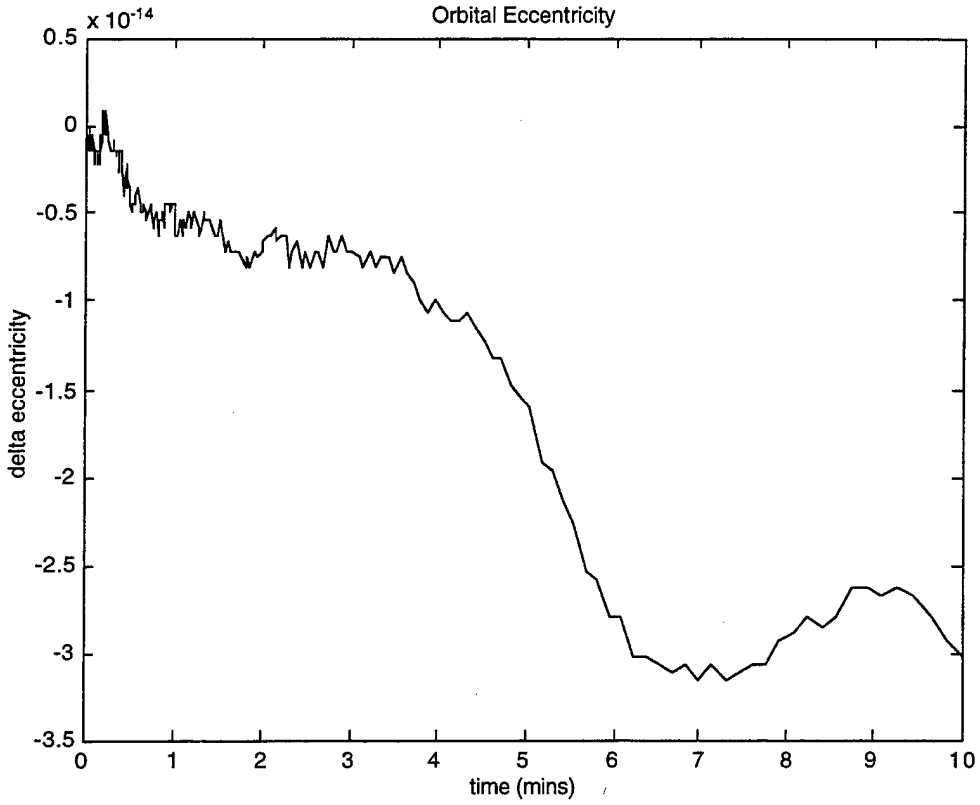

Figure 9: Δ Eccentricity

Different types of conic sections occur from two-body trajectories, depending on particle velocity and the spatial relations between the bodies. Multiple ACAPS simulations demonstrated the properties in Table 1 as part of the exoatmospheric validations.

Condition	Circle	Ellipse	Parabola	Hyperbola
Eccentricity	$e = 0$	$0 < e < 1$	$e = 1$	$e > 1$
Energy	$\varepsilon < 0$	$\varepsilon < 0$	$\varepsilon = 0$	$\varepsilon > 0$

Table 1: Conics Parameters

B. ENDOATMOSPHERIC

Endoatmospheric validations were based on the ballistic entry and lifting entry discussions by Hale, Ref. 2. Hale reduced the nonlinear ordinary differential equations of motion that describe a vehicle trajectory in the atmosphere to closed-form solutions, by making several "well-known" simplifications.

> The basic assumption for a ballistic (direct) entry is the absence of lift ($L = 0 = L/D = 0$). Additional assumptions and approximations include the neglect of the gravitational (and centrifugal) force during the early initial high-velocity phase of the entry trajectory where the drag force is so much larger than the gravitational force and the E/V [entry vehicle] is decelerating. Since it is the gravitational force that curves the trajectory as the E/V slows down and approaches the planetary surface, with a flat-Earth model the trajectory during the deceleration phase can be approximated by a straight line with a constant elevation angle ($\phi = \phi_{re}$).
>
> Implicit in the assumption that the lift is zero are the assumptions that the E/V is axisymmetric and that the angle of attack is zero (and remains equal to zero). Since an object in a trajectory maintains its initial attitude with respect to the inertial reference unless subjected to external forces, another implicit assumption is that the attitude of the E/V has somehow been appropriately adjusted prior to entry. [Ref. 2]

ACAPS Fidelity Degree 1 was selected for its non-rotating atmosphere to simulate the conditions described in Hale's Example 7-3-2. ACAPS takes into account the gravitational effects at high-velocity and numerically calculates the non-linear ordinary differential equations of motion without closed form simplification. So, some variation between ACAPS and Hale was expected. When Hale's Figure 7-3-3 and the ACAPS results given in Figure 10 were compared side by side, the results were virtually identical. Note on Figure 10 that the velocity axis is normalized to the velocity of re-entry, and that the ballistic coefficient is included to characterize each trajectory,

$$BC = \frac{Mg_o}{C_D A} \tag{98}$$

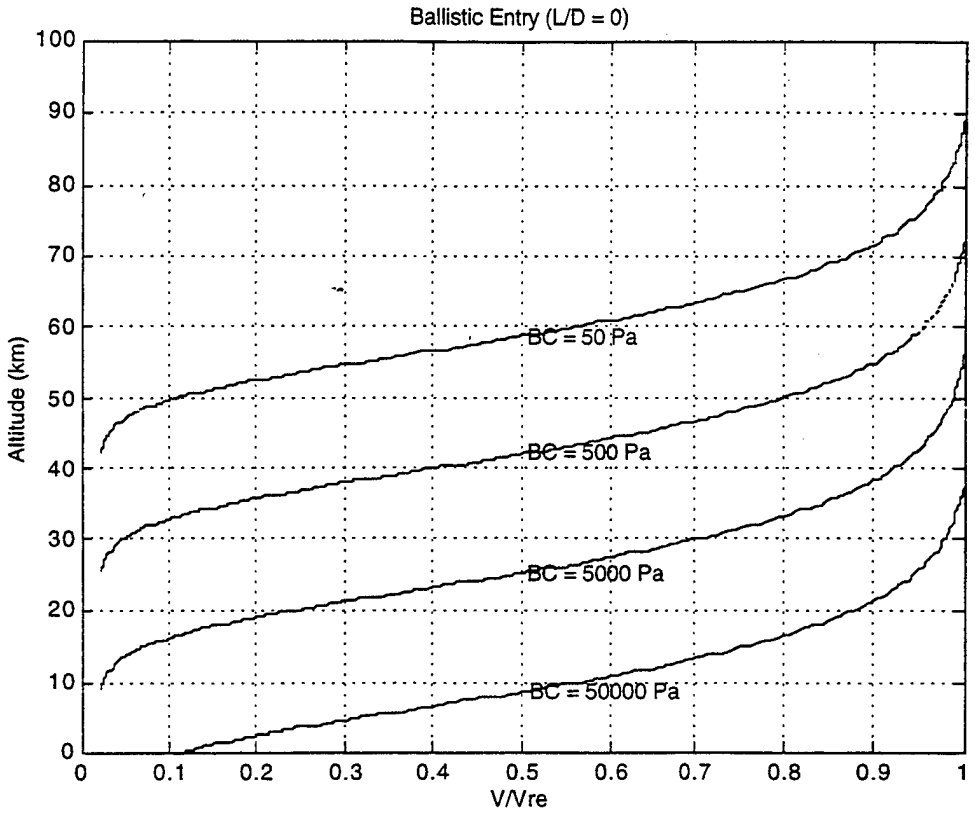

Figure 10: Ballistic Entry

The lifting entry is more dynamically robust than the ballistic entry. The entry corridor width is increased, acceleration on the vehicle is decreased, and control requirements are more flexible with increased maneuverability [Ref. 2]. The following validation demonstrates that ACAPS correctly models the lifting entry. $L \neq 0$ and $D \neq 0$ define a lifting entry. Hale made further assumptions for his lifting-entry study.

> An important lifting-entry trajectory is the *equilibrium glide*, which is a relatively flat glide in which the gravitational force is balanced by the combination of the lift and centrifugal forces. In addition to 'small' angle assumptions ($\sin\phi \cong 0$ and $\cos\phi \cong 0$) with respect to the elevation angle ϕ, it is further assumed that ϕ is changing slowly so that $d\phi/dt$ can be neglected (to a first approximation ϕ can be assumed to be constant) and that the lift-to-drag ratio is ≥ 0.5 or so.

ACAPS Fidelity Degree 1 was again selected for its non-rotating atmosphere to simulate the conditions described in Hale's Example 7-4-1. ACAPS was still expected to diverge slightly from Hale's solution, for reasons similar to those stated in the ballistic validation. However, a more exact measure was applied to contrast the two solutions. Hale's closed-form solutions and ACAPS' numerical solutions were plotted over one another to facilitate visual comparison, see Figure 11. The solutions were extremely close, so the ACAPS non-rotating atmosphere model was considered validated. Note in Figure 11 that the velocity axis is normalized to the circular orbit velocity, and that the lifting ballistic coefficient is included to characterize each trajectory,

$$LBC = \frac{BC}{L/D} \tag{99}$$

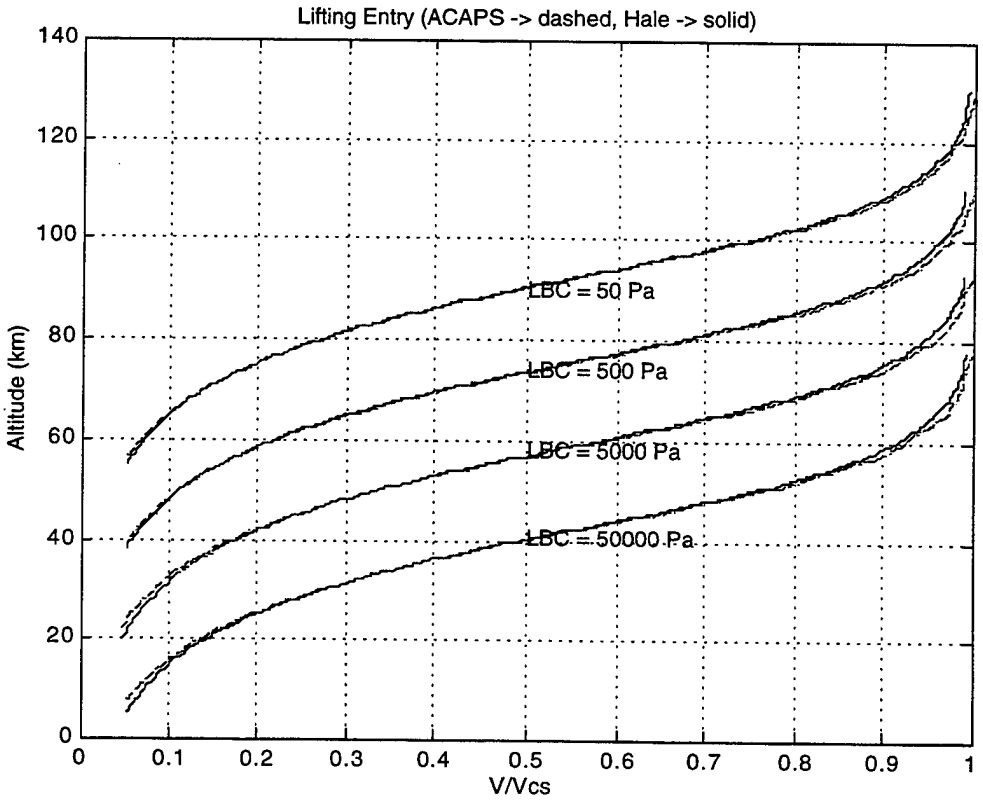

Figure 11: Lifting Entry

C. MARS MISSION 2001

The ACAPS rotating atmosphere model was validated with a comparison between the Mars 2001 Orbiter aerocapture behavior predicted by Jet Propulsion Laboratory [Ref. 4] and an ACAPS simulation of the same event. The Jet Propulsion Laboratory predictions apparently originated from one of NASA Langley's six-degrees-of-freedom Monte Carlo simulations. The parameters input to ACAPS for the Mars 2001 Orbiter simulation are listed below.

The planetary parameters were obtained from Ref. 3. The ACAPS equations of motion are configured to model body-fixed thrust by including the rocket equation as the seventh equation. However, thrust is zero in the Mars 2001 Orbiter simulation. Letting I_{sp} equal zero to reflect this condition puts a zero in the denominator of the rocket equation, Eq. (7), since,

$$v_e = I_{sp} g_o \tag{100}$$

The specific impulse, I_{sp}, was set to a default value of one second to avoid this division by zero singularity. The spacecraft model, initial state vector, and epoch parameters were obtained from Jet Propulsion Laboratory's Mars 2001 Design Team members.

```
PLANET
Radius, R                                = 3375.7 km
Gravitational constant, mu               = 4.282800e+13 m^3/sec^2
Stagnation point heating coefficient, C  = 3.55e-05
Gravitational coefficient, J2            = 0.0019605
Gravitational coefficient, J3            = 3.1449e-05

SPACECRAFT
Initial mass, M                          = 568.5 kg
Effective area related to CL and CD, A   = 5.52 m^2
Specific impulse, Isp                    = 1 s
Rocket exhaust velocity, ve              = 3.7584 m/s
Nose radius, Rn                          = 1 m
```

```
Stagnation point velocity exponent, M      = 3.15
Stagnation point density exponent, N       = 0.5

GUIDANCE
Coefficient of lift, CL                    = 0.3024
Coefficient of drag, CD                    = 1.68
L/D                                        = 0.18
Ballistic coefficient, BC                  = 61.3031 kg/m^2
Initial angle of bank, delta               = 0 deg
Initial angle of attack, alpha             = 0 deg

INITIAL STATE VECTOR
Initial radius, ro                         = 3500.7 km
Initial altitude, ht                       = 125 km
Initial latitude, phio                     = 92.92 degrees
Initial longitude, thetao                  = 186.9 degrees
Initial speed, vo                          = 6534 m/s
Initial flight path angle, gammao          = -10.91 degrees
Initial heading, psio                      = 20 degrees

EPOCH
Simulation time, tsim                      = 10 mins
Local mean solar time, lmst                = 1700
```

The guidance algorithm was lift-up, such that the flight path angle was set at atmospheric entry and no other guidance inputs directed the Orbiter thereafter.

The results were better than expected. ACAPS results, Figure 12 through Figure 15, matched the predictions from Jet Propulsion Laboratory, Figure 16, almost exactly. The ACAPS three-degrees-of-freedom rotating atmosphere model was considered validated. Note that the data outputs in the MATLAB Command Window and on the plots are chopped to three significant figures for consistency with the assessed total computational accuracy of the code.

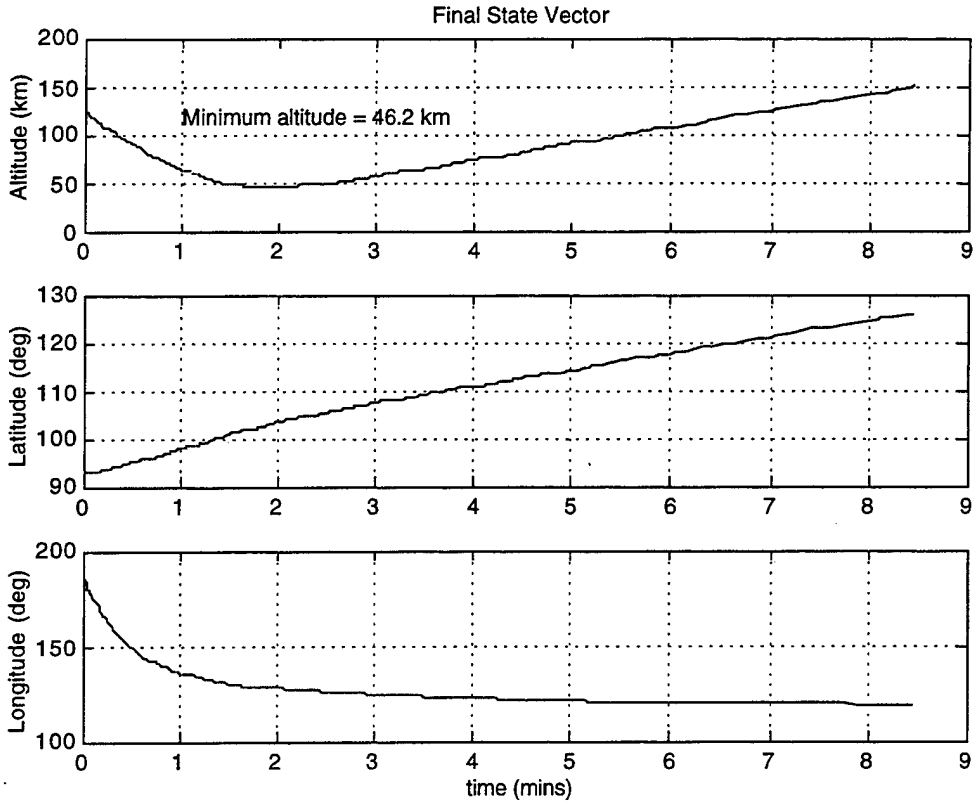

Figure 12: Mars 2001 Final State Vector

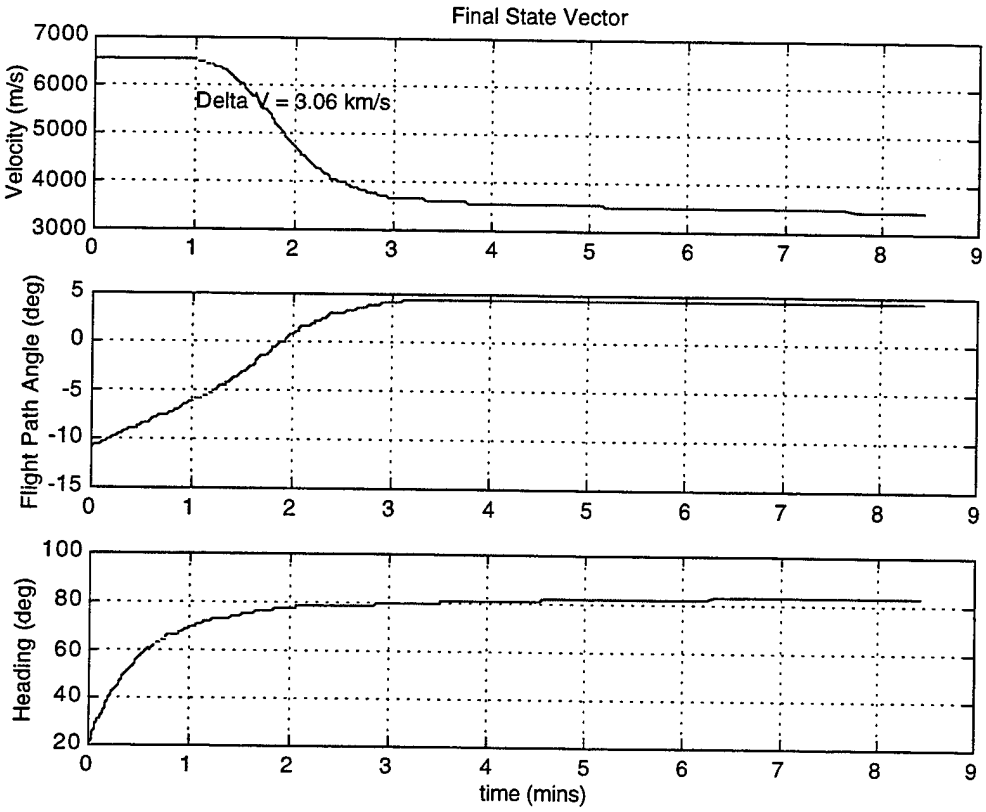

Figure 13: Mars 2001 Final State Vector

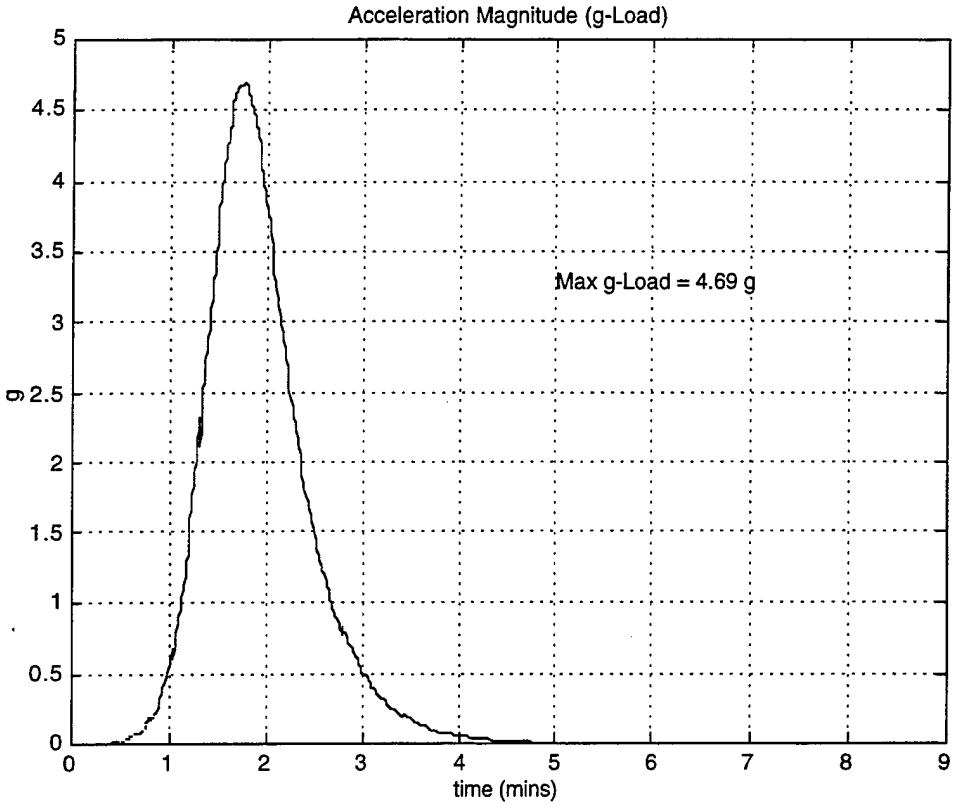

Figure 14: Mars 2001 Acceleration Magnitude

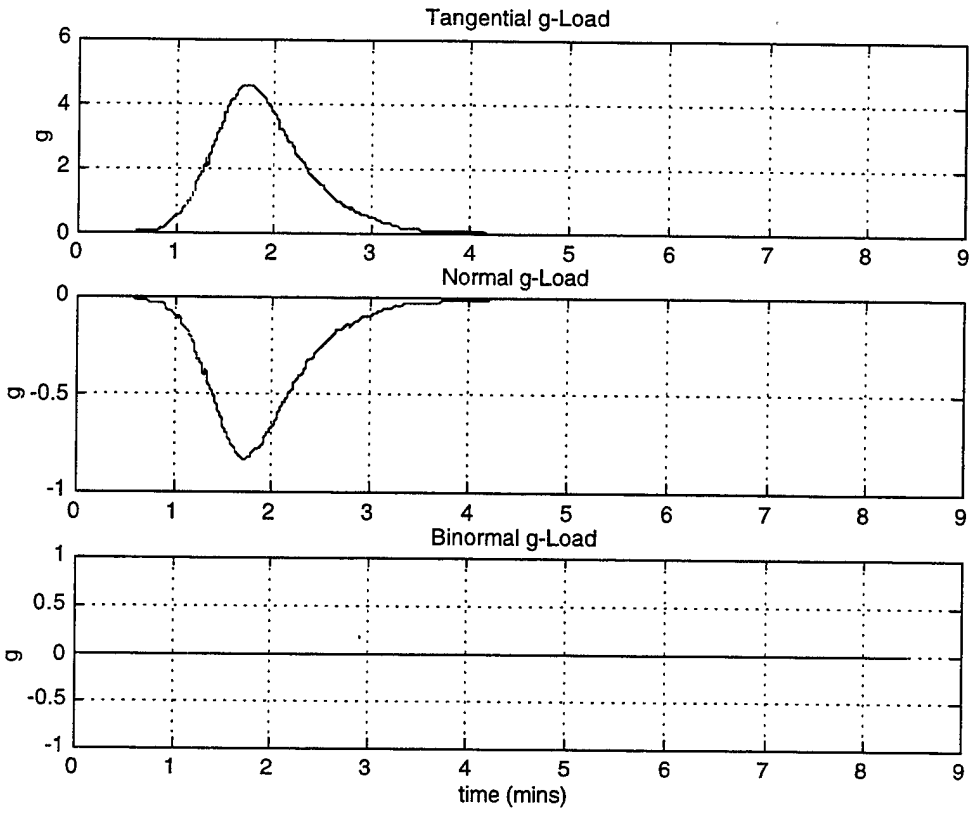

Figure 15: Mars 2001 Acceleration Elements

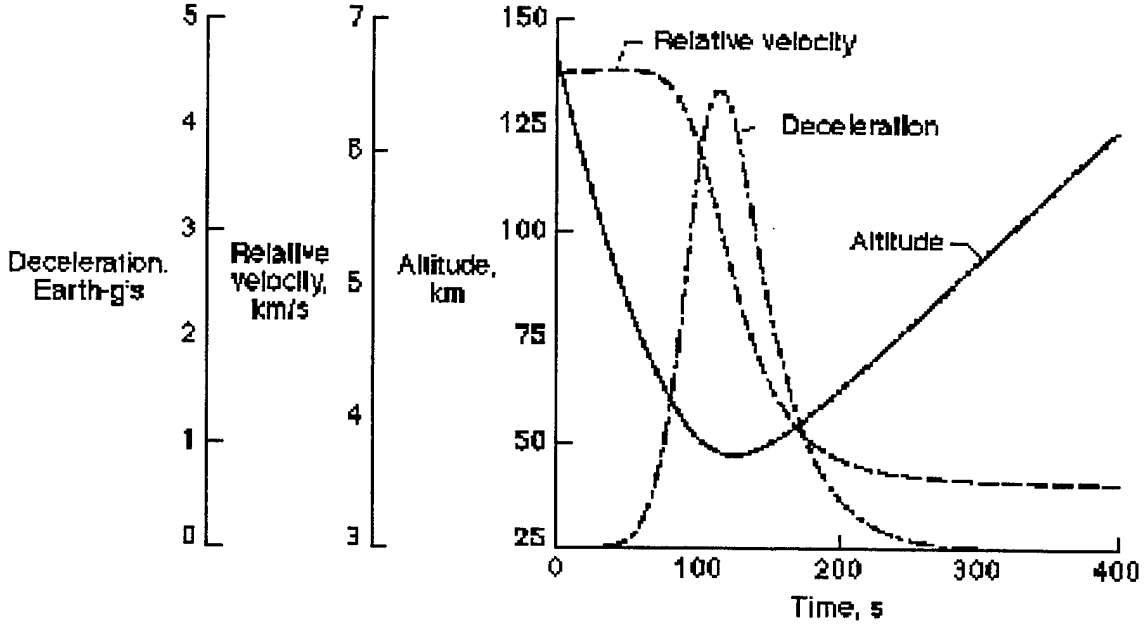

Figure 16: NASA Mars 2001 Orbiter Aerocapture Simulation

V. ANALYSIS OF MARS MISSION 2007 AEROCAPTURE

A. FINAL STATE VECTOR

An analysis of the Mars 2007 Orbiter aerocapture is conducted below to demonstrate the power of ACAPS. The spacecraft model used is the Mars 2001 Orbiter, for lack of a Mars 2007 Orbiter design to date. Figure 17 shows the 2001 Orbiter stowed in a biconic shell, surrounded by thermal protection shield. The biconic shape is designed so that rolling the spacecraft will modulate its lift vector to provide positive guidance control.

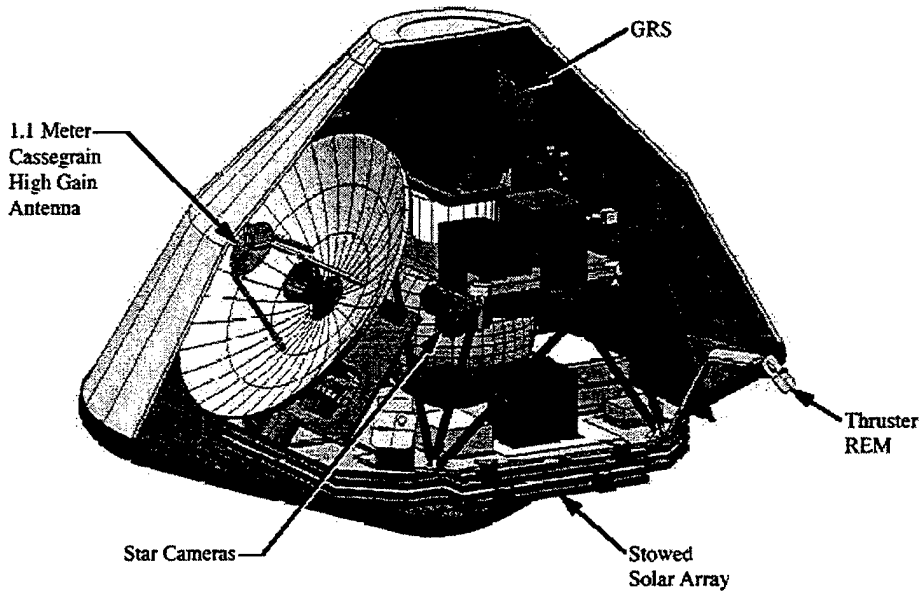

Figure 17: Mars 2001 Orbiter in Aerocapture Shell

The initial state vector was obtained from Zike's study of the Mars 2007 interplanetary trajectory that was based on Jet Propulsion Laboratory's MIDAS and CATO codes [Ref.11]. The planetary data was gathered from Ref. 3 and a MarsGRAM 3.7 example file that listed atmospheric density values from the 1976 Viking era. The

guidance was lift-up. Fidelity Degree 3 (look-up table atmospheric density, inverse-square gravity and rotating atmosphere) was selected. See the Command Window display below as a summary of input data,

```
PLANET
Radius, R                                  = 3375.7 km
Gravitational constant, mu                 = 4.282800e+13 m^3/sec^2
Stagnation point heating coefficient, C    = 3.55e-05
Gravitational coefficient, J2              = 0.0019605
Gravitational coefficient, J3              = 3.1449e-05

SPACECRAFT
Initial mass, M                            = 568.5 kg
Effective area related to CL and CD, A     = 5.52 m^2
Specific impulse, Isp                      = 1 s
Rocket exhaust velocity, ve                = 3.7584 m/s
Nose radius, Rn                            = 1 m
Stagnation point velocity exponent, M      = 3.15
Stagnation point density exponent, N       = 0.5

GUIDANCE
Coefficient of lift, CL                    = 0.3024
Coefficient of drag, CD                    = 1.68
L/D                                        = 0.18
Ballistic coefficient, BC                  = 61.3031 kg/m^2
Initial angle of bank, delta               = 0 deg
Initial angle of attack, alpha             = 0 deg

INITIAL STATE VECTOR
Initial radius, ro                         = 3500.4 km
Initial altitude, ht                       = 124.7 km
Initial latitude, phio                     = 33.195 degrees
Initial longitude, thetao                  = 288.85 degrees
Initial speed, vo                          = 6678 m/s
Initial flight path angle, gammao          = -10.48 degrees
Initial heading, psio                      = 180 degrees
```

```
EPOCH
Simulation time, tsim                    = 10 mins
Local mean solar time, lmst              = 1200
```

The simulation took approximately 15 seconds on a Macintosh G3 (233 MHz CPU, 66 MHz bus) Powerbook. The results are listed in the following figures. The altitude started at 125 km, Jet Propulsion Laboratory's official definition for the beginning of the Martian atmosphere, and bottomed out at 46.8 km, Figure 18. The time from atmospheric entry to exit was approximately 7 minutes.

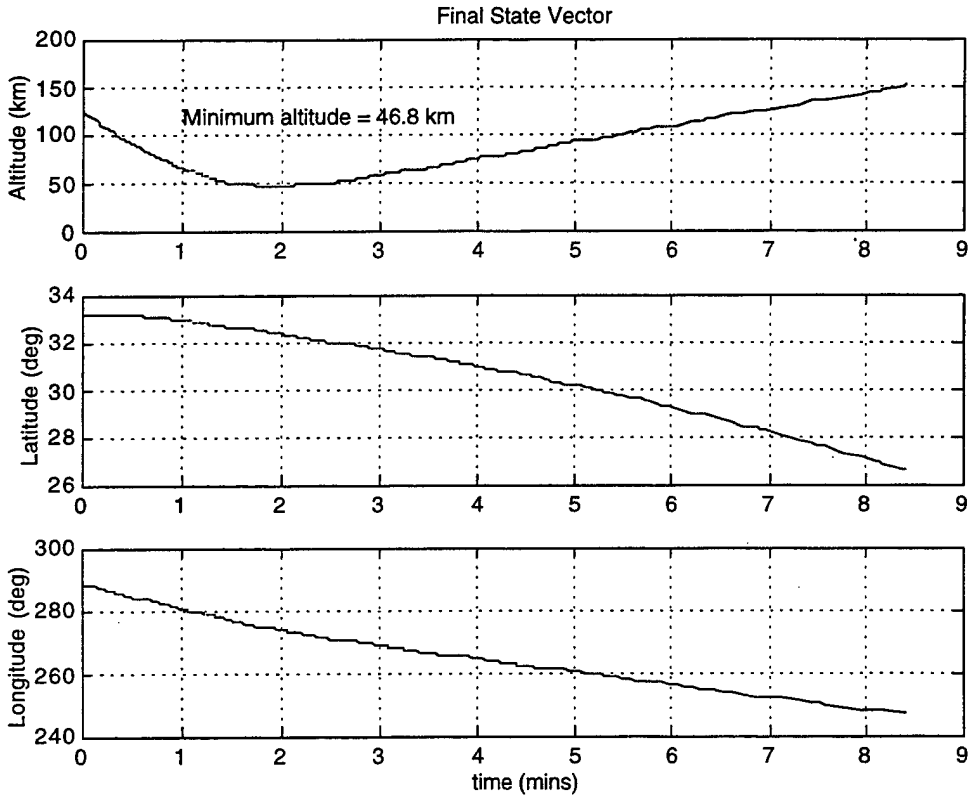

Figure 18: Mars 2007 Final State Vector

The initial velocity was 6678 m/s and decreased to approximately 3800 m/s, for a ΔV = 3.01 km/*s*, Figure 19. This could translate into significant mass savings at the launch pad, depending on the mass of the thermal protection shield.

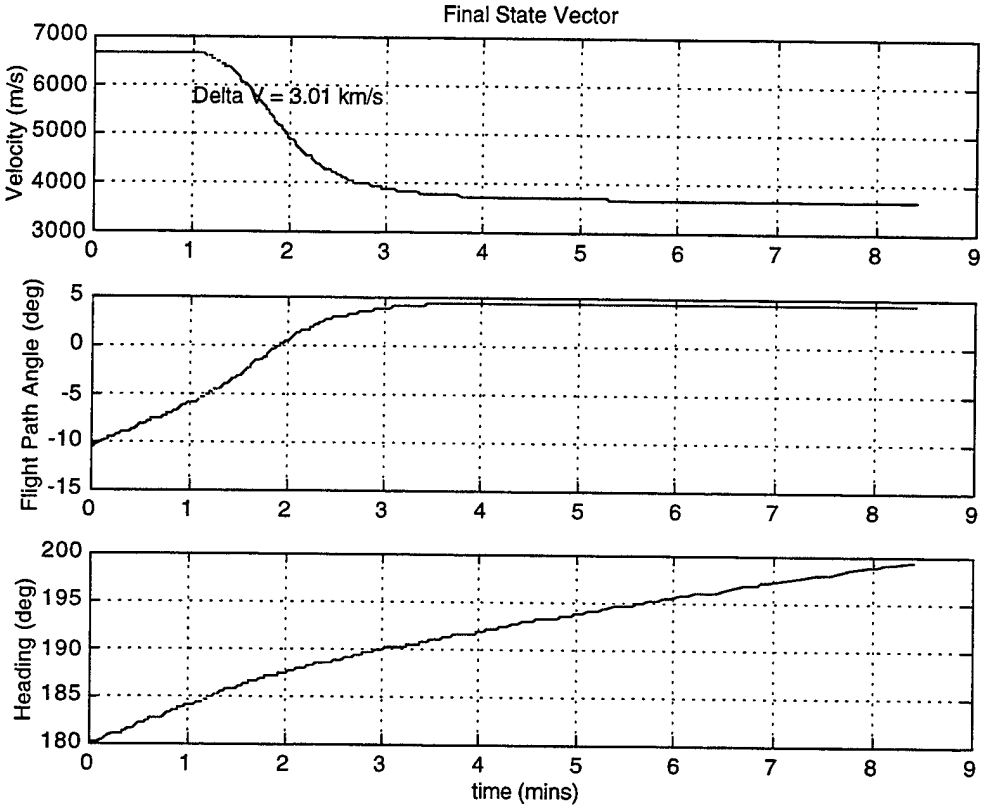

Figure 19: Mars 2007 Final State Vector

1. Visualization

The Visualize function in ACAPS is a very effective design tool. The simulated aerocapture trajectory is displayed over a three-dimensional planet. The user can move about the planet and zoom to view the trajectory from any perspective. See Figure 20 for a profile of the aerocapture trajectory.

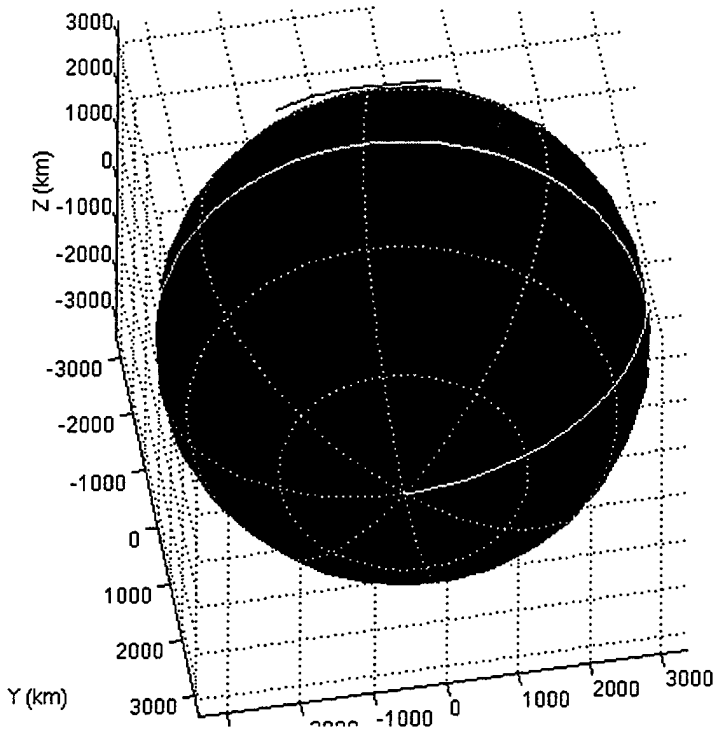

Figure 20: Mars 2007 Aerocapture Profile

A detailed planet graphic can be displayed over the planet's simple spherical representation, see Figure 21. Note that the simple sphere processes faster than the detailed planetary graphic, because of RAM and VRAM memory requirements.

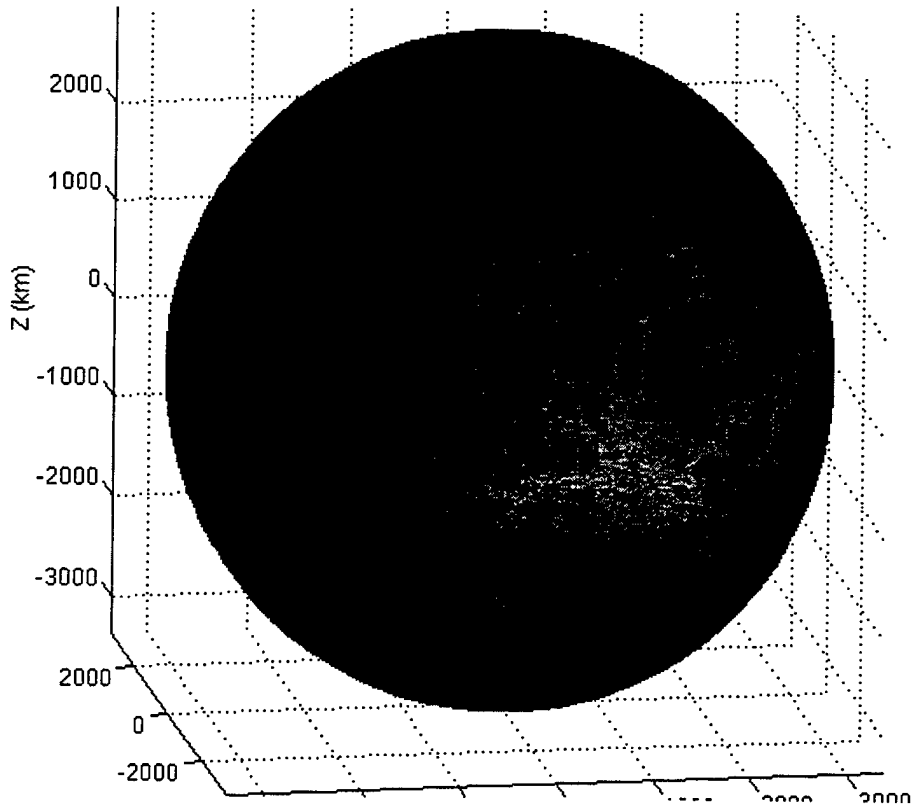

Figure 21: Mars 2007 Aerocapture Profile Planet Graphic

Payload engineers and scientists could gain insight into data gathering opportunities by studying the detailed planetary graphic to see where the spacecraft will aerocapture and start its first circularized orbit. The sun's terminator location could have considerable implications for possible control algorithm operations, if the vehicle were crossing from night into day during aerocapture where atmospheric diurnal variations might require large control adjustments. The light source about the planet represents the local mean solar time (LMST) at aerocapture. Note that the tilt of the planet is not included in the sunlight modeling.

2. Stagnation Point Heating

The stagnation point heating peaks at approximately 32.4 W/cm^2, a critical thermal protection shield design consideration, Figure 22.

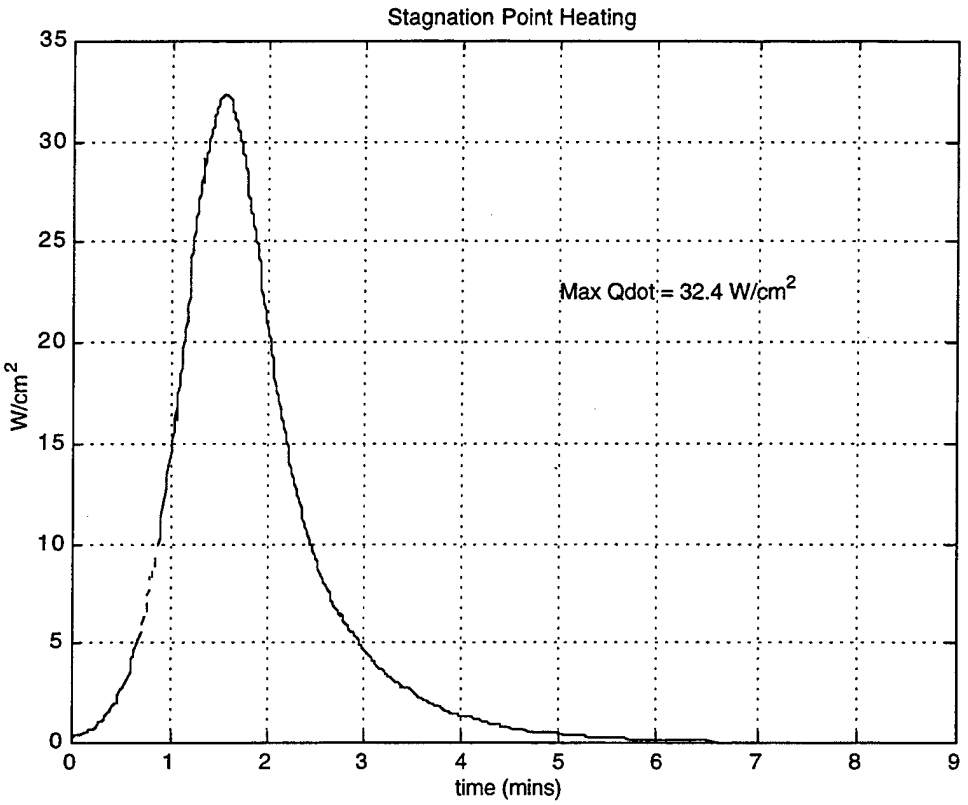

Figure 22: Mars 2007 Stagnation Point Heating

3. Accelerations

The maximum g-load is about 4.55 g's at just under two minutes into the aerocapture, Figure 23. The Human Mars Mission plans for the astronauts to withstand a maximum of 5 g's at aerocapture, after months of weightlessness prior to orbit insertion. Therefore, the predicted g-load for the human arrival is mission critical. The g-load is

further broken down into tangential, normal and binormal directions in Figure 24. Exactly what type of accelerations the vehicle sustains is important for equipment load design.

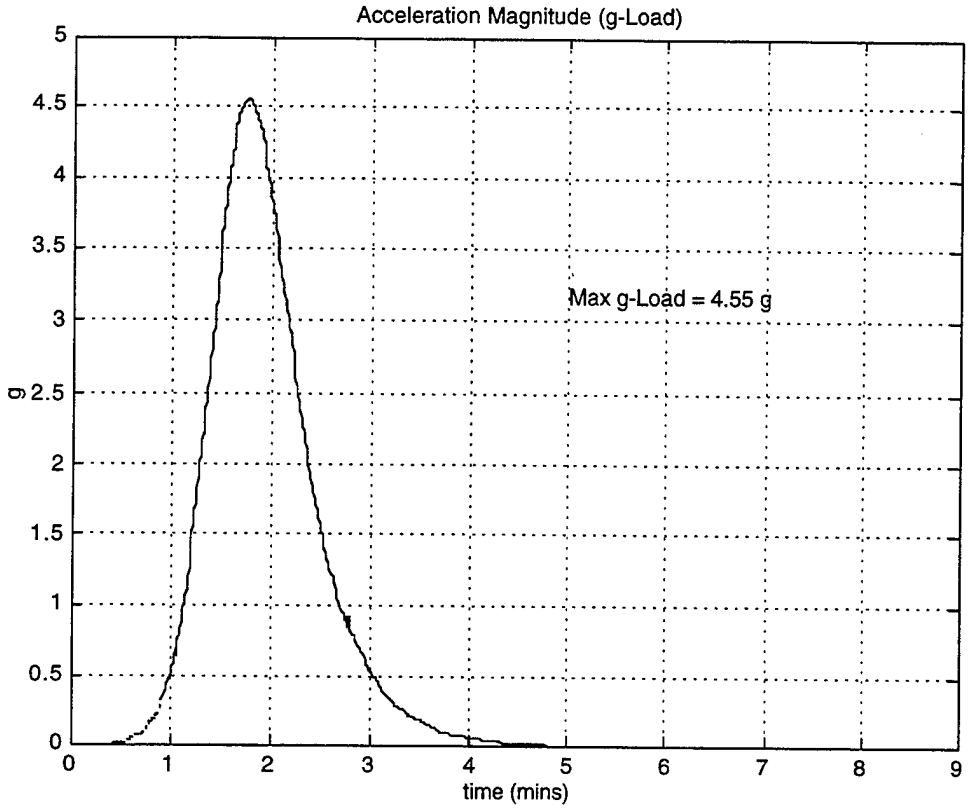

Figure 23: Mars 2007 Acceleration Magnitude

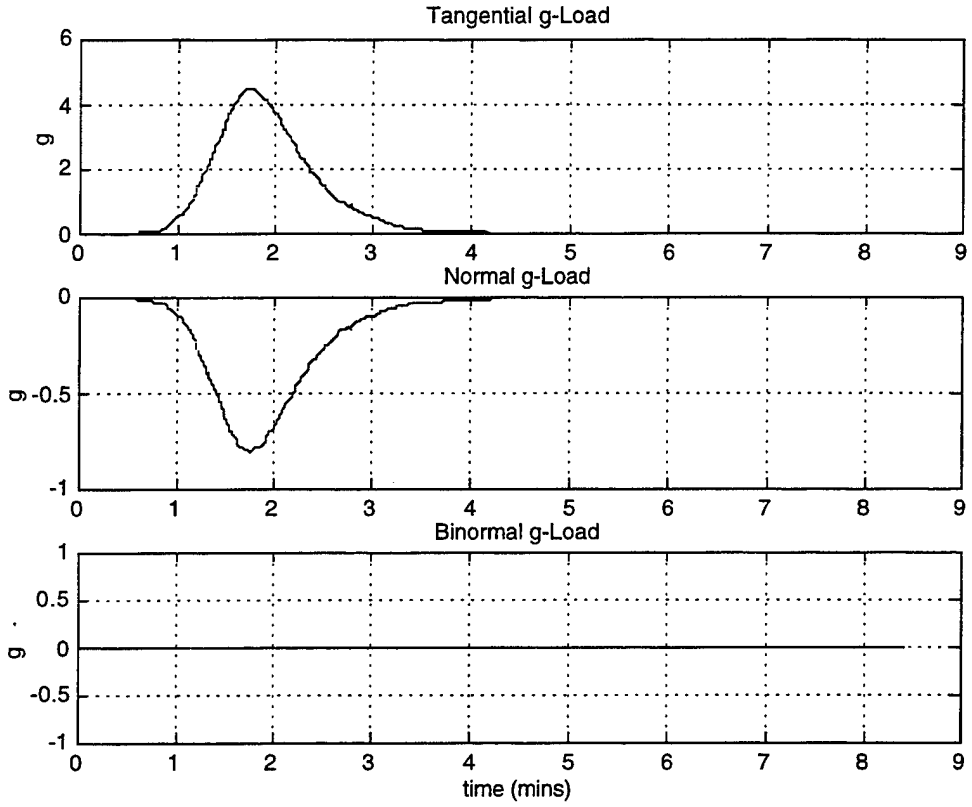

Figure 24: Mars 2007 Acceleration Elements

B. ORBITAL ANALYSIS

The Orbit function maps the final state vector to apoapse, calculates the ΔV required to circularize the orbit, plots the circularized orbit for one period, and calculates its orbital elements.

1. Visualization

An apoapse burn immediately after aerocapture raises the periapse to the final circularized orbit. The ΔV input at apoapse for this maneuver is simulated and the succeeding circular orbit trajectory is projected by the ACAPS Orbit function. The

following is displayed in the Command Window: the trajectory type; the circularized orbital period; the classical orbital elements of the circularized orbit; the apoapse altitude after aerocapture; the time to apoapse burn from atmospheric entry at 125 km; the ΔV gained by aerocapture; the ΔV required to circularize the orbit; and the total ΔV for the aerocapture. See listing below for the output displayed in the Command Window for the Mars 2007 Mission. The entire scenario is automatically visualized in three dimensions, Figure 25.

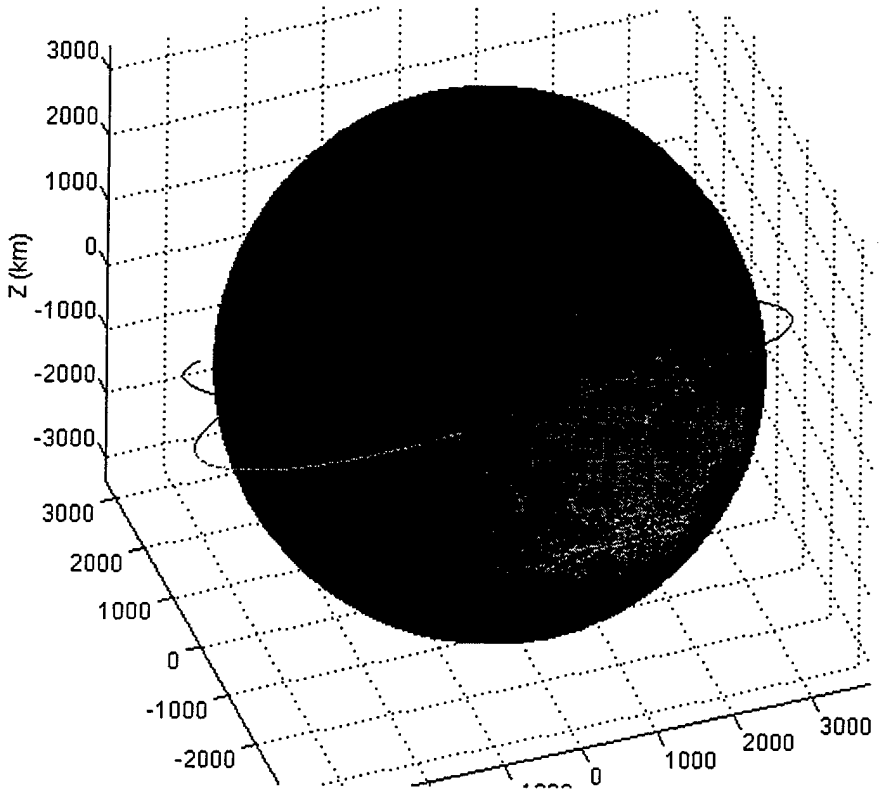

Figure 25: Mars 2007 Aerocapture Visualization

2. Classical Elements

ELLIPTICAL TRAJECTORY

Period	= 104 mins
Period	= 1.73 hrs

NOTE: Selecting exponential atmosphere for high-altitude modeling.

Semimajor Axis	= 3480 km
Eccentricity	= 0.0809
Inclination	= 147 degrees
Longitude of Ascending Node	= 17.2 degrees
Argument of Perigee	= 21.1 degrees

3. APOAPSE BURN AND ΔV

Apoapse altitude	= 385 km
Apoapse burn time from entry	= 33 mins
Delta V gained by aerocapture	= 3.01 km/s
Delta V required at apoapse	= 139 m/s
Total delta V gained	= 2.87 km/s

C. CORRIDOR DEFINITION

The Corridor function calculates the two flight path angles that define the aerocapture corridor. The user inputs the maximum acceptable stagnation point heating and g-load; the minimum permissible apoapse altitude; the desired flight path angle accuracy; and an initial flight path angle approximation. These inputs constrain and direct the Corridor function algorithm as it searches for the corridor bounds. The upper

dynamic is defined as the shallowest initial flight path angle where the vehicle trajectory after aerocapture is elliptical. The lower dynamic is defined as the steepest initial flight path angle with which the vehicle does not fly below the minimum desired apoapse altitude, or exceed the maximum stagnation point heating and g-load limits. The upper and lower dynamics are found for both lift-up and lift-down guidance. The upper bound of the aerocapture corridor is defined by the lift-down upper dynamic; and the lower bound of the aerocapture corridor is defined by the lift-up lower dynamic.

The aerocapture corridor for the Mars 2007 Mission was calculated by the ACAPS Corridor function and presented in Figure 26. The inputs selected in the Corridor GUI were the following: 100 W/cm^2 maximum stagnation point heating; 10 g maximum g-load; 125 km minimum apoapse; $\gamma = -10^\circ$ mean initial flight path angle; and resultant flight path angle accuracy to the hundredths. APPENDIX A lists the associated Command Window display. This listing gives the lift-up and lift-down flight path angles to achieve the apoapse altitude window, given the spacecraft position and velocity. In summary, the corridor is 1.67° wide, with the upper bound set by the lift-down upper dynamic at $\gamma = -8.96^\circ$, and the lower bound set by the lift-up lower dynamic at $\gamma = -10.63^\circ$. Figure 26 plots both the lift-up and lift-down guidance trajectories with their respective lower and upper dynamics. The aerocapture corridor is useful because it defines some limits on control design.

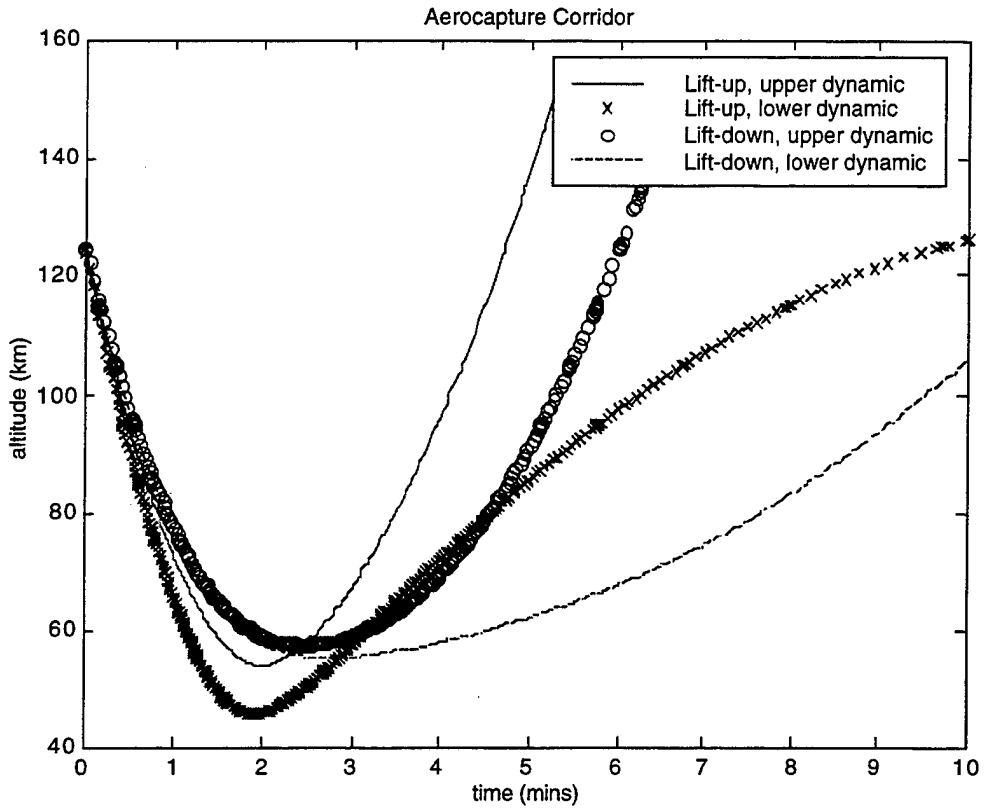

Figure 26: Mars 2007 Aerocapture Corridor

D. FLIGHT PATH ANGLE DETERMINATION

The Gamma function searches for the range of flight path angles with which the vehicle aerocaptures into a nominal apoapse altitude band. This band is defined by the user as the minimum and maximum apoapse altitudes immediately after aerocapture. This reflects the circular orbit altitude that results after the apoapse burn. The user also enters the maximum stagnation point heating limit; the maximum g-load limit; the desired flight path angle accuracy; and an initial flight path angle approximation. Flight path angles attempted by the Gamma function algorithm to meet these constraints are listed in the Command Window with respective overshoot or undershoot comments.

The inputs selected for the Gamma GUI for the Mars 2007 aerocapture were the following: 375 km to 425 km apoapse altitude window; 100 W/cm^2 maximum stagnation

point heating; 10 g maximum g-load; $\gamma = -10^{\circ}$ mean initial flight path angle; and a resultant flight path angle accuracy to the hundredths. These inputs constrained the Gamma function algorithm as it searched for the target flight path angles.

The algorithm starts with the initial flight path angle input by the user. Then it "searches" for the nominal flight path angles as follows: it simulates the aerocapture; projects the apoapse altitude after exiting the atmosphere; and evaluates that result against the user-defined apoapse altitude window. If the apoapse is within the target altitude limits, the flight path angle is output to the Command Window. If the apoapse is outside the target altitude limits, then the algorithm adjusts the flight path angle per lift-up or lift-down control characteristics and continues the search until converging upon an answer. If the algorithm is unable to meet the specified criteria, the closest answers are displayed in the Command Window.

APPENDIX B lists the associated Command Window display. This listing gives the lift-up and lift-down flight path angles to achieve the apoapse altitude window, given the spacecraft position and velocity. In summary, the Mars 2007 Orbiter lift-up flight path angle which achieved capture within the predefined apoapse window was $\gamma = -10.48^{\circ}$. The lift-down flight path angle was undetermined. The user selected flight path angle accuracy of hundredths was too large, and the algorithm was unable to converge within limits. The Command Window output indicated that $\gamma = -9.04^{\circ}$ yielded a 60.2 km apoapse and $\gamma = -9.03^{\circ}$ yielded a 1290 km apoapse. This aerocapture flight path angle determination is useful because it defines some limits on the approach from interplanetary trajectory.

E. REQUIRED FIDELITY

The Compare function runs two simulations with different degrees of fidelity, then plots their final state vectors together to contrast accuracies. The higher fidelity is

considered the more accurate standard. The respective run times are listed in the Command Window. The user can weigh the run times against accuracies to help determine which fidelity model will suffice for their purposes.

The Mars 2001 comparison in the previous Validation section showed that ACAPS' and Jet Propulsion Laboratory's 2001 Orbiter projections were very close. The following Compare function plots study the fidelity options in detail and reveal specifically why fidelity degree 2 or 3 is probably the most that will ever be required for first-cut trade studies.

Figure 27 and Figure 28 compare fidelities 1 and 5, the lowest and highest options. Fidelity degree 1 uses the following models: a global exponential atmospheric density; an inverse-square gravity; and a non-rotating atmosphere. Fidelity degree 5 uses the following models: a look-up table atmospheric density; a J_3 gravitational model; and a rotating atmosphere. There is quite a bit of variation between the final state vectors, and the simulation time for the higher fidelity model is over three times as long as the lower fidelity.

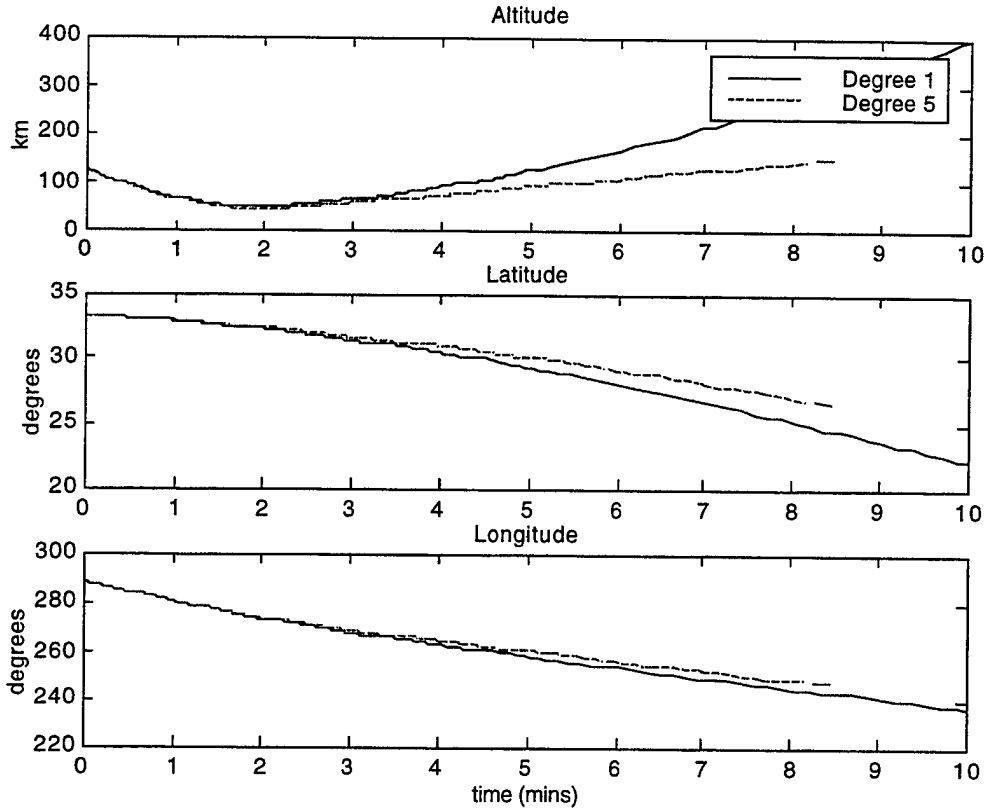

Figure 27: Degrees 1 and 5 Fidelity Comparison

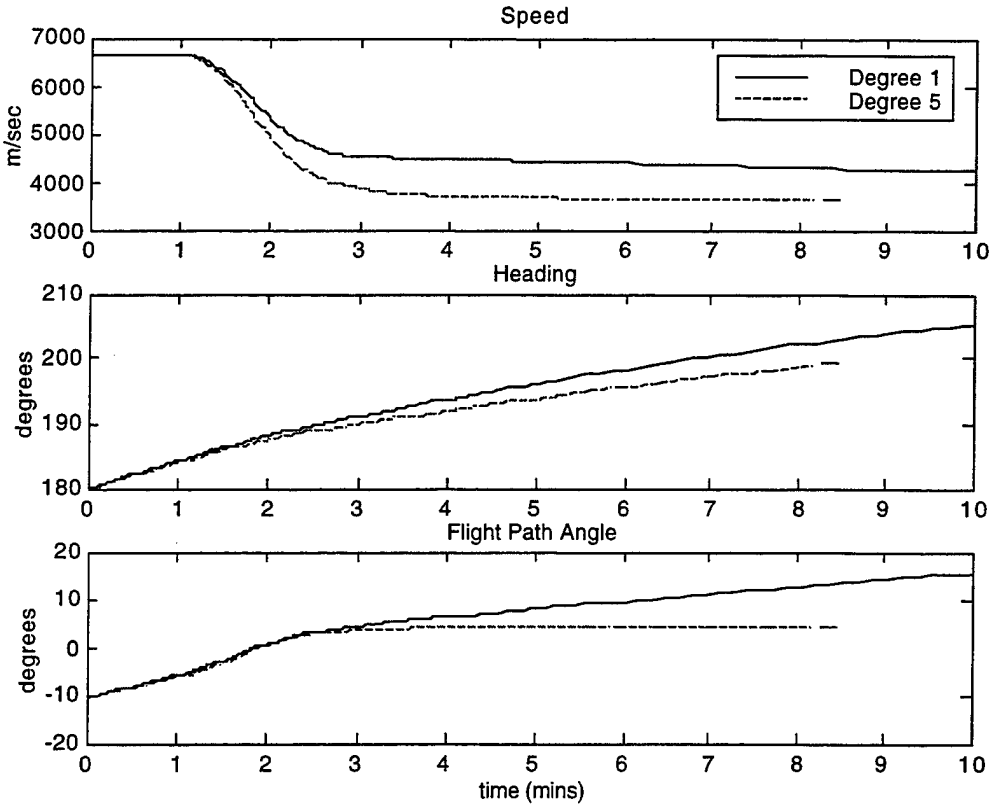

Figure 28: Degrees 1 and 5 Fidelity Comparison

```
Degree 1 simulation time:
11.9 sec

Degree 5 simulation time:
38.2 sec
```

Figure 29 and Figure 30 compare fidelity degrees 2 and 5. Fidelity degree 2 uses the following models: a global exponential atmospheric density; an inverse-square gravity; and a rotating atmosphere. The difference between final state vectors is almost indistinguishable. Since the simulation conditions between fidelity degrees 1 and 2 are only different by the atmospheric rotation, we conclude that atmospheric rotation is necessary for a simulation accurate enough for a first-cut trade study. The global exponential atmospheric density apparently works at this level of analysis almost as well as the look-up table. The difference in run times is still over twice as long for the higher fidelity model. These times reflect simulation on a G3 233MHz Powerbook.

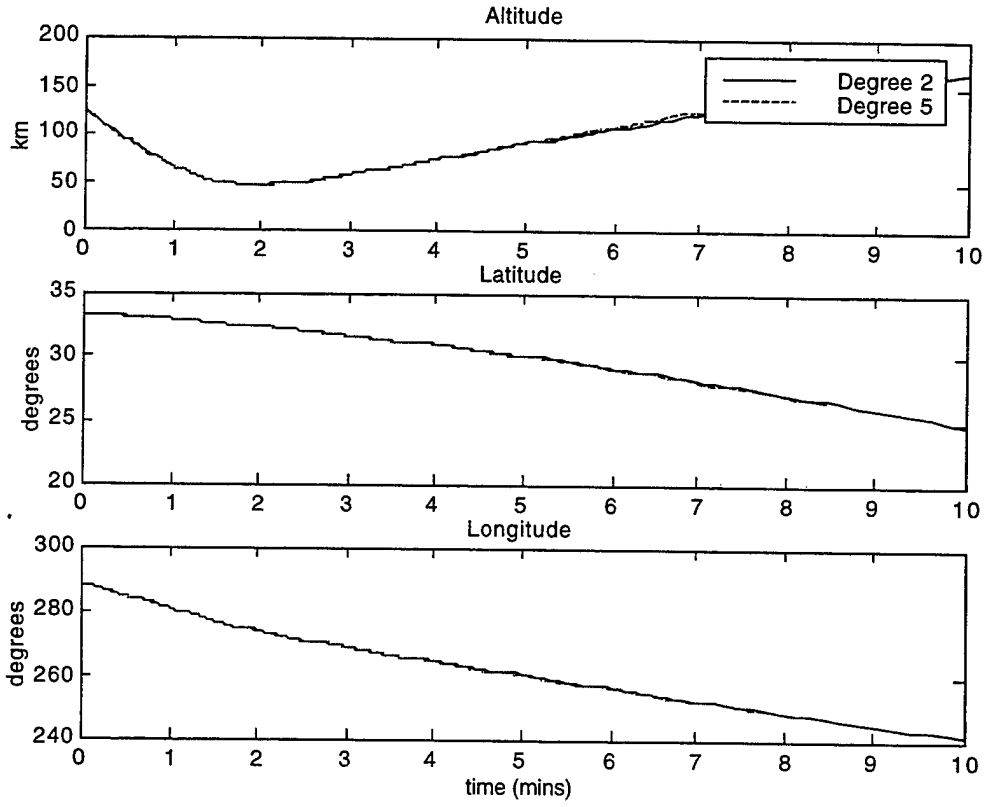

Figure 29: Degrees 2 and 5 Fidelity Comparison

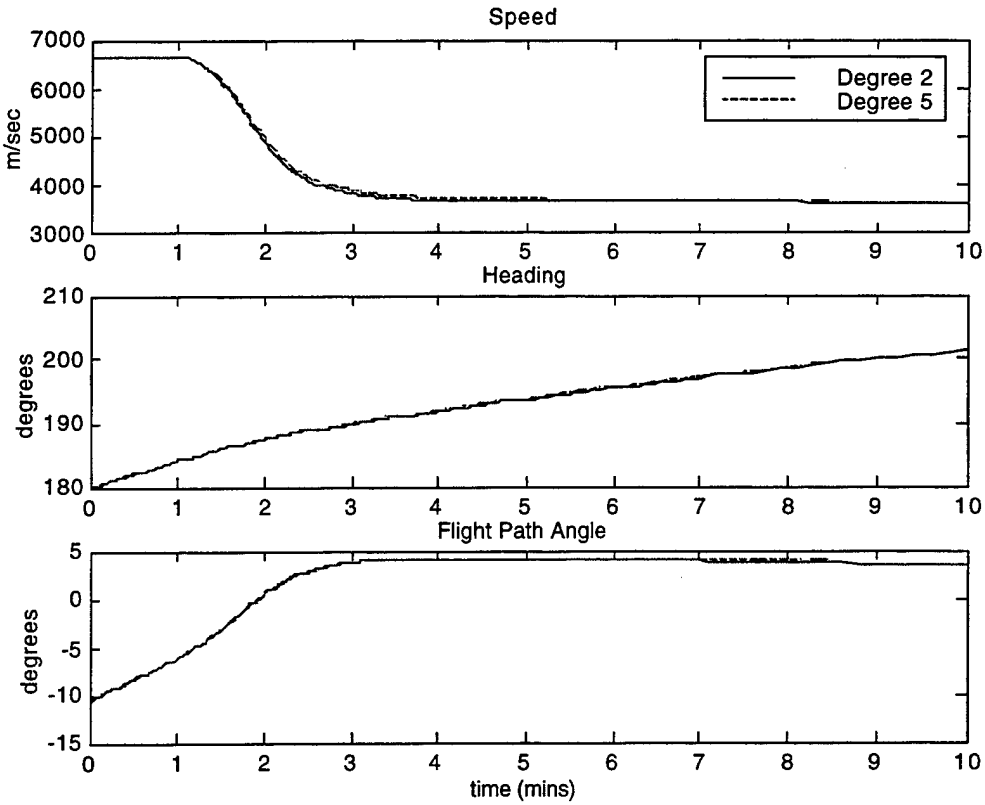

Figure 30: Degrees 2 and 5 Fidelity Comparison

```
Degree 2 simulation time:
16.2 sec

Degree 5 simulation time:
38.2 sec
```

VI. SUMMARY AND RECOMMENDATIONS

A. ACAPS

Future space missions will require aerocapture. In the near future there is the Mars 2005 Sample Return Mission, the Mars 2007 Mission, the Neptune/Triton Mission, and the Human Mars Mission. ACAPS is a validated, fast, user-friendly, first-cut mission design tool developed for the teams who will plan these missions. Currently, no such tool exists in the JPL design team's inventory. ACAPS will provide insights into aerocapture missions that will help JPL make more informed decisions faster, better and cheaper. ACAPS was developed for mission upgradeability with state-of-the-art software, making it cost effective and adaptable.

B. FUTURE RESEARCH

The ACAPS atmosphere model could be upgraded for higher fidelity simulations, for applications such as targeting an entry landing zone, or aerobraking studies. These models could also include the relationship between the vehicle lift and Mach number, which would improve the dynamics modeling.

A guidance block that was designed to access an M-file for control inputs was pre-installed in the ACAPS SIMULINK block diagram models. The Guidance block is currently inactive, but an M-file that commanded inputs could be integrated for aerocapture guidance algorithm development.

The equations of motion include the rocket equation as the seventh equation, which models body-fixed thrust. This capability is currently inactive, but a constant or time dependent thrust assigned in the spacecraft model M-file could be used for aerocapture optimization through body-fixed thrusting.

The dynamics do not currently model lag times from input commands to the actual maneuver or propagate errors from such lag times. However, by modifying the appropriate M-files, such error propagation could be modeled and possibly employed in hypersonic waverider control algorithm development. This high fidelity control model could also be used to study synergetic maneuvers.

Performance databases exist for several types of aerocapture thermal protection shields. These databases could be reduced to characteristic curves, fit with polynomials, and input into ACAPS. ACAPS could compare this thermal protection shield database against the stagnation point heating output from an aerocapture simulation to determine first-cut thermal protection shield mass requirements. ACAPS could then compare this thermal protection shield mass with a propellant mass estimate, based on the ΔV output from the Orbit function and a user-defined propulsion system, to facilitate a first-cut trade study of mission feasibility.

VII. ACAPS USER'S MANUAL

A. ACAPS OVERVIEW

1. Description

ACAPS is a mission design tool intended as a first-cut estimate of final state vectors, related g-loading and stagnation point heating. ACAPS was originally developed for aerocapture in the Mars 2007 Mission. However, aerobraking, direct entry (re-entry) and aerogravity assist are possible upgrades. ACAPS is coded in MATLAB 5.2, with SIMULINK. ACAPS users interact with intuitive graphic user interfaces (GUIs) that enable a fast learning curve. Modular development enhances mission upgradeability.

2. Equations of Motion

Equations of motion for a spacecraft in three-degrees-of-freedom (3-DOF) mathematically describe the aerocapture event. These non-linear, forward propagating, ordinary differential equations (ODE) incorporate planetary constants, spacecraft characteristics, and the initial state vector with atmospheric and gravitational models of the user's choice. The ODEs are solved numerically in SIMULINK. All parameters are entered in SI (MKS) units, then converted into non-dimensional, canonical units for calculation. This allows more efficient numerical solution for faster and more accurate results. Output is converted back into dimensional SI units.

3. MATLAB

ACAPS is coded in MATLAB 5.2. Many M-files, also referred to here as modules, work together to accomplish ACAPS tasking. M-files are grouped lines of

MATLAB code that serve a particular purpose. The M-files reside in the ACAPS Toolbox, which contains the *Design, GUI, Guidance, Modeling, Output, Simulation, Support* and *Validation* Toolboxes. They are transparent to the mission designer, who interacts exclusively with the GUIs. Modifications to planets, spacecraft, guidance algorithms, atmospheric or gravitational models is possible.

ACAPS code is platform independent and fully transportable between UNIX workstations and Macintosh or PC computers, as long as MATLAB version 5.2, or possibly a later version, is installed. ACAPS is best transferred between Mac and PC platforms as a compressed file to avoid media incompatibilities. ACAPS incorporates state-of-the-art functions only available on MATLAB 5.2, and will not run on prior MATLAB versions.

4. SIMULINK

SIMULINK is a MATLAB Toolbox that works in a simulation environment. The SIMULINK model file *aeras7_blk.mdl* runs the ACAPS numerical solver and incorporates a Guidance block into the loop. The numerical ODE solver is a highly efficient ODE 45, Dormand-Prince, variable step method. It is an explicit Runge-Kutta (4,5) formula pair. Results are retrieved from the loop and output to the MATLAB Workspace.

The user can monitor non-dimensional, canonical simulation time, *tsimbar*, by calling up *aeras7_blk.mdl*. To do this, type *aeras7_blk* in the MATLAB Command Window, or select it with the open file option in the Command Window Toolbar. When doing the latter, *All Files* must be selected to view *aeras7_blk.mdl*, since it is not an M-file. Once viewing *aeras7_blk*, double-click on the clock symbol in the top right. Monitoring time this way slows the simulation slightly.

5. Installation

Copy the ACAPS Toolbox into the computer hard drive. Start MATLAB and go into the MATLAB Path option. Select all of the ACAPS Toolboxes one at a time and append them into the MATLAB Path and then Save. MATLAB will now recognize the M-files in the ACAPS Toolbox from the MATLAB Workspace. Set MATLAB Memory Requirements Preferred Size to 48M to handle planet graphics.

When using FTP to put the ACAPS Toolbox into a UNIX environment, make the following adjustment. Start MATLAB and go into the MATLAB Path option. Select the ACAPS Toolboxes one at a time and append them into the MATLAB Path and Save. MATLAB will now recognize the M-files in the ACAPS Toolbox from the MATLAB Workspace. Open the GUI Toolbox and delete the *acapsvis.mat.bin* and *plottergui.mat.bin* files, if they exist. They are unusable artifacts from the FTP. After you delete these files, replace them by typing *save acapsvis* and *save plottergui* in the MATLAB Workspace to recreate the *_.mat* files necessary for MATLAB to build the GUIs.

6. Getting Started

Initialize ACAPS by typing *acaps* in the MATLAB Command Window. A GUI will appear. You will continue to interact with the GUI and Command Window throughout the simulation. The left column of pop-up menu buttons is input ("Flight Plan"); below that are the data review, simulation, reset and clear push buttons. The center column is mission design ("Design"), and the right column is output ("Post-flight").

Sequence your input selections from top to bottom, left to right. Change a particular entry by repressing that button before simulation. The *Data* button displays all selected inputs in the Command Window and ensures all necessary data is entered

before simulation. The *Reset* button is for use only after a simulation, followed by a "Post-flight" option. ONLY press *Clear* to erase ALL data from the MATLAB Workspace. This is appropriate after any "Design" option.

B. GRAPHIC USER INTERFACE (GUI)

1. Flight Plan

Spacecraft Select the spacecraft. The following spacecraft parameters are entered into the MATLAB Workspace, via the M-file *spacecraft.m*:

initial spacecraft mass, M (kg)
reference surface area, A (m^2)
quadratic drag polar coefficient, K
coefficient for stagnation point heating, C (S.I. Units)
atmospheric density power for stagnation point heating, N_{stag}
velocity power for stagnation point heating, M_{stag}
specific impulse of the body fixed thrust, I_{sp} (s)

State Vector Select the initial state vector components, referenced to IORB flag 42, central body equator and meridian of epoch. The following data is entered into the MATLAB Workspace, via the M-file *init.m*:

initial radius from planet center is calculated, r_o = height + R (m)
initial latitude in the planet reference frame, ϕ_o (degrees)
initial longitude in the planet reference frame, θ_o (degrees)
initial speed of the spacecraft relative to the planet-fixed frame, v_o (m/s)
initial flight path angle of the spacecraft, γ_o (deg)

initial heading angle of the spacecraft, ψ_o (degrees)

Planet Select the planet. The following planetary constants are entered into the MATLAB Workspace, via the M-file *planet.m*:

polar or equatorial radius as applicable, R (m)

planetary gravitational constant, μ (m^3/s^2)

average distance from the sun, (AU)

angular velocity of the planet, Ω (rad/s)

J_2 gravitational parameter

J_3 gravitational parameter

Guidance Select the guidance module for aerocapture. The following data is entered into the MATLAB Workspace, via the M-file *guidance.m*:

initial coefficient of lift, C_L

initial coefficient of drag, C_D

thrust over time, T (N)

angle of attack over time, α (degrees)

angle of bank over time, δ (degrees)

Time Select the aerocapture simulation time. The following data is entered into the MATLAB Workspace, via the M-file *simtime.m*:

simulation time, t_{sim} (minutes)

local mean solar time of aerocapture event, lmst (24 hour clock)

Fidelity Select the simulation fidelity. The following options relate to selectable degrees, via the M-file *fidelity.m*:

Fidelity	Atmospheric Density	Gravity	Atmospheric Rotation
1	Exponential	Inverse-square	Non-rotating
2	Exponential	Inverse-square	Rotating
3	Look-up Table	Inverse-square	Rotating
4	Look-up Table	J_2	Rotating
5	Look-up Table	J_3	Rotating

Table 2: Fidelity Degrees

2. Preflight

Data Review the applicable simulation data. The following data is listed in the MATLAB Workspace, via the M-file *list.m*:

radius, R (m)

planetary gravitational constant, μ (m^3/s^2)

spacecraft initial mass, m (kg)

reference surface area, A (m^2)

specific impulse of body fixed thrust, I_{sp} (s)

rocket exhaust velocity, v_e (m/s)

coefficient for stagnation point heating, C (W s $m^{1/2}$/kg)

atmospheric density power for stagnation point heating, N_{stag}

velocity power for stagnation point heating, M_{stag}

initial coefficient of lift, C_L

initial coefficient of drag, C_D

initial angle of band, δ (deg)

initial angle of attack, α (deg)

initial radius from planet center is calculated, $r_o = ht + R$ (m)

initial altitude from surface, ht (m)

initial latitude in the planet reference frame, ϕ_o (degrees)

initial longitude in the planet reference frame, θ_o (degrees)

initial velocity in the spacecraft reference frame, v_o (m/s)

initial flight path angle in the spacecraft reference frame, γ_o (deg)

initial heading angle in the spacecraft reference frame, ψ_o (degrees)

simulation time, t_{sim} (minutes)

local mean solar time of epoch, lmst (24 hour clock)

3. Flight

Simulate Simulate aerocapture by numerical solution of the equations of motion. Choose the atmosphere and gravity models by selecting fidelity. The higher the fidelity of the model, the more accurate the results, but longer the simulation time. The lowest fidelity model is an exponential atmospheric density, rotating atmosphere, and inverse-square gravity. The M-file *grho7_sims.m* manages the simulation.

Reset Select only after a "Post-flight" option. Resets initial conditions and clears propagated vectors in the Command Workspace. DO NOT select after a "Design" option, since these options reset the initial conditions to something other than that previously selected in the "Flight Plan" section.

Clear Erases all data in the MATLAB Workspace. Select after all "Design" options.

Help General information. Type an M-file's name in the MATLAB Command Window without the *.m* suffix to read its description. All M-files are grouped by function in the Toolboxes.

4. Postflight

Visualize Call up the visual GUI for trajectory plotting and animation, via the M-file *visualize.m*.

Select Plot User plots select variables from simulation, via the M-file plotter.m. These variables include the following:

time (min)

altitude (km)

latitude ϕ (deg)

longitude θ (deg)

v (m/s)

flight path angle γ (deg)

flight path heading ψ (deg)

thrust (N)

angle of attack α (deg)

angle of bank δ (deg)

stagnation point heating (W/cm^2)

tangential acceleration (g)

normal acceleration (g)

binormal acceleration (g)

magnitude of acceleration (g)

lift (N)

drag (N)

angular momentum X-component (kg m^2/s)

angular momentum Y-component (kg m^2/s)

angular momentum Z-component (kg m^2/s)

angular momentum magnitude (kg m^2/s)

atmospheric density (kg/m^3)

radial gravitational force (m/s^2)

List Data List resultant simulation data within a defined range, via the M-file listdata.m. This data includes the following:

r (m)

latitude ϕ (deg)

longitude θ (deg)

v (m/s)

flight path angle γ (deg)

flight path azimuth ψ (deg)

stagnation point heating (W/cm^2)

magnitude of acceleration (g)

State Plot the final state vector components, via the M-file *plotsv.m*. The minimum altitude (m) is annotated on the r plot; the ΔV is annotated on the v plot. The ΔV is the change in velocity from entry to exit.

r (m)

latitude ϕ (deg)

longitude θ (deg)

v (m/s)

flight path angle γ (deg)

flight path azimuth ψ (deg)

Thermal Plot spacecraft stagnation point heating (W/cm^2) over time, via the M-file *thermal.m*.

G-Load Plot the g-loading (g's) over time in the tangential, normal and binormal (lateral) directions and g-load magnitude, via the M-file *gload.m*.

Controls Plot the following guidance parameters over time, via the M-file *guidvec.m*: thrust, T (N); angle of attack, α (degrees); and angle of bank, δ (degrees).

5. Design

Orbit Select only after simulation. Propagates the orbit after aerocapture event and displays the trajectory type, period, classical orbital elements, time to burn, ΔV saved during the aerocapture, ΔV for the apoapse burn and total ΔV in the Command Window, via the M-file *plotorbit.m*.

Corridor Select after "Flight Plan" selection, and without prior simulation.

This module calls the "Corridor" GUI, via the M-file *corridorgui.m*, where maximum stagnation point heating, maximum g-load acceleration, minimum desired apoapse altitude and desired flight path angle accuracy are entered to the Command Workspace. When *Search* is selected, the module seeks the flight path angles that define the aerocapture corridor.

Gamma Select after "Flight Plan" selection.

Determine the required flight path angle to meet a nominal apoapse altitude or final circular science orbit. Input to the "Gamma" GUI, via the M-file *ballisticgui.m*, the maximum and minimum apoapse altitudes, maximum stagnation point heating, maximum g-load and desired flight path angle accuracy. If the control limits are exceeded during simulation, the user is afforded the option of continuing or stopping the search for the optimal flight path angle.

Compare Select after "Flight Plan" selection, and without prior simulation. Compares two simulations' fidelities and run times, via the M-file *compare.m*.

6. Visualize

Trajectory Plot the 3-D trajectory over the planet graphic, via the M-file *plot3d.m*.

Animate Plot the 3-D trajectory over the planet graphic sequentially to see direction of travel, via the M-file *animate.m*. *Trajectory* must have been run prior to animation for (*xx, yy, zz*) coordinate matrix and axis.

Rotate On Enable the built-in MATLAB graphics rotation function.

Rotate Off Disable the built-in MATLAB graphics rotation function.

Axes Toggle on/off the planet-fixed axes.

Grid Toggle on/off the built-in MATLAB graphics grid function.

Lat/Long Toggle on/off the latitude/longitude lines. Managed by the M-file latlong.m.

Light Toggle on/off a "room light" to see the planet on the dark side. Managed by the M-file lightcont.m.

5 deg Select 5 degree increments for the Up/Down and Right/Left buttons, via the M-file *fivedeg.m*.

45 deg Select 45 degree increments for the Up/Down and Right/Left buttons, via the M-file *fourfivedeg.m*.

Up Move the viewer's perspective up by the increment amount selected via the M-file *moveup.m* (5 degrees is the default).

Down Move the viewer's perspective down by the increment amount selected, via the M-file *movedown.m* (5 degrees is the default).

Right Move the viewer's perspective right by the increment amount selected, via the M-*file moveright.m* (5 degrees is the default).

Left Move the viewer's perspective left by the increment amount selected, via the M-file *moveleft.m* (5 degrees is the default).

Zoom In Move the viewer's perspective in by 50 percent, via the M-file *zoomin.m.*

Zoom Out Move the viewer's perspective out by 50 percent, via the M-file *zoomout.m.*

Target Center and zoom in by 1.5 times on a selected object, via the M-File *pan2d.m.* Crosshairs appear on the graphic after *Target* selection. Bullseye these crosshairs on the object and click the mouse button.

Refresh Redraw the planet graphic, via the M-file *planetemp.m. Trajectory* or *Animate* in the output button column must be selected afterwards to redraw the aerocapture flight path.

Sphere Select a sphere for the planet. Much faster response time than the planet graphic. Managed by the M-file plansph.m.

Graphic Select the graphic for the planet. Much slower response time than the simpler planet sphere. Managed by the M-file plangraph.m.

C. MODEL MODIFICATIONS

1. General

ACAPS' modular design is intended for relatively easy upgradeability and modification. Aerocapture simulations are possible about any planet, with any spacecraft, any guidance algorithm, and any fidelity atmosphere or gravity model by altering the applicable M-file modules described below.

2. Planets

A planet may be added in the M-file *planet.m*. Follow the format of Mars and provide appropriate planetary constants. Update the selections prompt at the beginning of the module.

3. Spacecraft

Additional spacecraft designs may be added by duplicating M-file *scmod2001.m* and providing appropriate parameters. Update the selections prompt in the module *spacecraft.m* to include the new module.

4. Guidance

Additional guidance algorithms may be added by following the format in module *clcd2001.m*. However, since guidance algorithms are relatively diverse and sometimes complex, guidance modules may take on entirely different structures. Update the selections prompt in the M-file *guidance.m* to include the new module.

5. Atmosphere

High fidelity atmosphere models are complex, and most current models are in FORTRAN. Rather than backing out a MATLAB code from FORTRAN, it is recommended that a look-up table, equations or algorithm that generated the FORTRAN subroutine be used to build a new MATLAB atmosphere model.

Set a user prompt in the M-file *grho7sims.m* that selects multiple fidelity models. Running the appropriate aeras7_eom.m module makes this selection. The lowest fidelity module is *aeras71eom.m*, which calls function *gbar.m* and *rhobar.m*. The *bar* suffix indicates these calls are made during non-dimensional calculations.

Aerocapture is only a 10-20 minute evolution. Despite the best atmospheric models available, unpredictable errors arise from a combination of the following sources: diurnal variations, 27-day solar-rotation cycle, 11-year cycle of Sun spots, semi-annual variations, cyclical variations, rotating atmosphere and winds [Ref. 9]. The spacecraft control system must be robust enough to autonomously react to unanticipated atmosphere changes. In lieu of this, the lowest fidelity exponential atmosphere model may realistically yield an acceptable estimate of final state vector for a first-cut design tool.

6. Gravity

Similar to the atmospheric model discussion above, the inverse-square gravity model may realistically yield an acceptable estimate of final state vector for a first-cut design tool. If not, a module may be built to accommodate higher fidelity models for the higher harmonics.

7. Output

Any quantity calculable from the final state vector (r, θ, ϕ, v, ψ, γ), or resultant g, ρ, L and D is programmable as an output, i.e. stagnation point heating (ρ, v) in the M-file *thermal.m*.

D. EXTRAS

1. Start/Pause

Start, Stop or Pause a simulation by opening the SIMULINK file *aeras7_blk.mdl* and selecting *Simulation* on the top bar, then select Start, Stop or Pause. Open any SIMULINK file by typing its name without the *.mdl* suffix in the MATLAB Command Window.

2. Discontinuities

ACAPS computations are undefined at $\phi = 90°$, $v = 0$ m/s, and $\gamma = 90°$. This is a result of division by zero in the equations of motion. A geosynchronous velocity singularity exists in fidelity degrees 2 through 5, the planet-fixed coordinate frame propagators. If the spacecraft inertial velocity nears $v = 0.24$ (km/s), then the relative velocity to the surface of the planet nears numerical zero, which results in a singularity. ACAPS recognizes this singularity and will flag the user with a Warning in the Command Window.

APPENDIX A: CORRIDOR FUNCTION OUTPUT

```
-10 deg flight path angle overshoots minimum altitude.
-10 deg flight path angle yields elliptical trajectory.

-9 deg flight path angle overshoots minimum altitude.
-9 deg flight path angle yields hyperbolic trajectory.

-9.9 deg flight path angle overshoots minimum altitude.
-9.9 deg flight path angle yields elliptical trajectory.

-9.8 deg flight path angle overshoots minimum altitude.
-9.8 deg flight path angle yields elliptical trajectory.

-9.7 deg flight path angle overshoots minimum altitude.
-9.7 deg flight path angle yields elliptical trajectory.

-9.6 deg flight path angle overshoots minimum altitude.
-9.6 deg flight path angle yields hyperbolic trajectory.

-9.69 deg flight path angle overshoots minimum altitude.
-9.69 deg flight path angle yields elliptical trajectory.

-9.68 deg flight path angle overshoots minimum altitude.
-9.68 deg flight path angle yields elliptical trajectory.

-9.67 deg flight path angle overshoots minimum altitude.
-9.67 deg flight path angle yields elliptical trajectory.

-9.66 deg flight path angle overshoots minimum altitude.
-9.66 deg flight path angle yields elliptical trajectory.

-9.65 deg flight path angle overshoots minimum altitude.
-9.65 deg flight path angle yields elliptical trajectory.

-9.64 deg flight path angle overshoots minimum altitude.
-9.64 deg flight path angle yields elliptical trajectory.

-9.63 deg flight path angle overshoots minimum altitude.
-9.63 deg flight path angle yields hyperbolic trajectory.

Lift-up     upper dynamic     = -9.64     deg

-10 deg flight path angle overshoots minimum altitude.

-11 deg flight path angle undershoots minimum altitude.

-10.1 deg flight path angle overshoots minimum altitude.

-10.2 deg flight path angle overshoots minimum altitude.

-10.3 deg flight path angle overshoots minimum altitude.

-10.4 deg flight path angle overshoots minimum altitude.

-10.5 deg flight path angle overshoots minimum altitude.
```

-10.6 deg flight path angle overshoots minimum altitude.

-10.7 deg flight path angle undershoots minimum altitude.

-10.61 deg flight path angle overshoots minimum altitude.

-10.62 deg flight path angle overshoots minimum altitude.

-10.63 deg flight path angle overshoots minimum altitude.

-10.64 deg flight path angle undershoots minimum altitude.

Lift-up apoapse: minimum altitude has exceeded the Control Value.

Lift-up lower dynamic = -10.63 deg

Inverting angle of bank.

-10 deg flight path angle undershoots minimum altitude.
-10 deg flight path angle yields elliptical trajectory.

-9 deg flight path angle overshoots minimum altitude.
-9 deg flight path angle yields elliptical trajectory.

-8 deg flight path angle overshoots minimum altitude.
-8 deg flight path angle yields hyperbolic trajectory.

-8.9 deg flight path angle overshoots minimum altitude.
-8.9 deg flight path angle yields hyperbolic trajectory.

-8.99 deg flight path angle overshoots minimum altitude.
-8.99 deg flight path angle yields elliptical trajectory.

-8.98 deg flight path angle overshoots minimum altitude.
-8.98 deg flight path angle yields elliptical trajectory.

-8.97 deg flight path angle overshoots minimum altitude.
-8.97 deg flight path angle yields elliptical trajectory.

-8.96 deg flight path angle overshoots minimum altitude.
-8.96 deg flight path angle yields elliptical trajectory.

-8.95 deg flight path angle overshoots minimum altitude.
-8.95 deg flight path angle yields hyperbolic trajectory.

Lift-down upper dynamic = -8.96 deg

-10 deg flight path angle undershoots minimum altitude.

-9 deg flight path angle overshoots minimum altitude.

-10 deg flight path angle undershoots minimum altitude.

-9.1 deg flight path angle undershoots minimum altitude.

-9.01 deg flight path angle overshoots minimum altitude.

-9.02 deg flight path angle overshoots minimum altitude.

-9.03 deg flight path angle overshoots minimum altitude.

-9.04 deg flight path angle undershoots minimum altitude.

Lift-down apoapse: minimum altitude has exceeded the Control Value.

Lift-down	lower dynamic	= -9.03	deg
Corridor width		= 1.67	deg

APPENDIX B: GAMMA FUNCTION OUTPUT

-10 deg flight path angle overshoots minimum altitude.

-11 deg flight path angle undershoots minimum altitude.

-10 deg flight path angle overshoots minimum altitude.

-10.9 deg flight path angle undershoots minimum altitude.

-10.8 deg flight path angle undershoots minimum altitude.

-10.7 deg flight path angle undershoots minimum altitude.

-10.6 deg flight path angle undershoots minimum altitude.

-10.5 deg flight path angle undershoots minimum altitude.

-10.4 deg flight path angle overshoots minimum altitude.

-10.5 deg flight path angle undershoots minimum altitude.

-10.49 deg flight path angle undershoots minimum altitude.

-10.48 deg flight path angle overshoots minimum altitude.

Flight path angle = -10.48 deg
Apoapse altitude = 385 km

Inverting angle of bank.

-10 deg flight path angle undershoots minimum altitude.

-9 deg flight path angle overshoots minimum altitude.

-10 deg flight path angle undershoots minimum altitude.

-9.9 deg flight path angle undershoots minimum altitude.

-9.8 deg flight path angle undershoots minimum altitude.

-9.7 deg flight path angle undershoots minimum altitude.

-9.6 deg flight path angle undershoots minimum altitude.

-9.5 deg flight path angle undershoots minimum altitude.

-9.4 deg flight path angle undershoots minimum altitude.

-9.3 deg flight path angle undershoots minimum altitude.

-9.2 deg flight path angle undershoots minimum altitude.

-9.1 deg flight path angle undershoots minimum altitude.

-9 deg flight path angle overshoots minimum altitude.

-9.1 deg flight path angle undershoots minimum altitude.

-9.09 deg flight path angle undershoots minimum altitude.

-9.08 deg flight path angle undershoots minimum altitude.

-9.07 deg flight path angle undershoots minimum altitude.

-9.06 deg flight path angle undershoots minimum altitude.

-9.05 deg flight path angle undershoots minimum altitude.

-9.04 deg flight path angle undershoots minimum altitude.

-9.03 deg flight path angle overshoots minimum altitude.

-9.04 deg flight path angle undershoots minimum altitude.

Flight path angle converged to range -9.04 and -9.03 deg.
Apoapse altitude converged to range 60.2 and 1290 km.

LIST OF REFERENCES

1 Allen, G.A., Gage, P.J., Venkatapathy, E., Olynick, D.R., and Wercinski, P.F., "A Web-based Analysis System for Planetary Entry Vehicle Design", *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, American Institute of Aeronautics and Astronautics Paper 98-4826, September 1998.

2 Hale, Francis J., *Introduction to Space Flight*, Prentice-Hall, Inc., Englewood Cliffs, 1994, pp 196-244.

3 *Mars Observer Planetary Constants and Models*, Jet Propulsion Laboratory D-3444, November 1990.

4 *Mars Surveyor Program 2001 Orbiter Mission, Proposal Information Package,* Jet Propulsion Laboratory, Pasadena, June 1997.

5 *Move up to MATLAB 5*, The MathWorks, Inc., Natick, August 1998.

6 *Proceedings of NASA Jet Propulsion Laboratory's Workshop on Aeroassisted Technologies*, Pasadena, California, 1997.

7 Ross, I. Michael, *3-DOF Model: Canonical EOMs Notes*, Monterey, February 1998.

8 *Using MATLAB*, The MathWorks, Inc., Natick, 1997, pp 8-2 to 8-46.

9 Vallado, David A., *Fundamentals of Astrodynamics and Applications*, Mcgraw-Hill Companies, Inc., New York, 1997, pp. 490-510.

10 Vinh, N.X., Busemann, A., and Culp, R., *Hypersonic and Planetary Flight Mechanics*, Univ. of Michigan Press, Ann Arbor, Michigan, 1980, pp. 100-107.

11 Zike, Steve, *Mission Analysis for the Mars 2007 Opportunity*, Master's Thesis, Naval Postgraduate School, Monterey, California, December 1998.

BIBLIOGRAPHY

Allen, G.A., Gage, P.J., Venkatapathy, E., Olynick, D.R., and Wercinski, P.F., "A Web-based Analysis System for Planetary Entry Vehicle Design", *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, American Institute of Aeronautics and Astronautics Paper 98-4826, September 1998.

Boyce, W.E., and DiPrima, R.C., *Elementary Differential Equations and Boundary Value Problems*, Fifth Edition, John Wiley and Sons, Inc., New York, 1992.

Braun, R.D., Powell, R.W., and Hartung, L.C., "Effect of Interplanetary Trajectory Options on a Manned Mars Aerobrake Configuration," National Aeronautics and Space Administration TP-3019, August 1990.

Chapman, D.R., "An Analysis of the Corridor and Guidance Requirements for Supercircular Entry into Planetary Atmospheres," National Advisory Committee for Aeronautics TR R-55, 1960.

Desai, P.N., and Braun, R.D., "Mars Parking Orbit Selection," *The Journal of Astronautical Sciences*, Volume 39, Number 4, 1991, pp. 447-467.

Gerald, C.F., and Wheatley, P.O., *Applied Numerical Analysis*, Fifth Edition, Addision-Wesley Publishing Company, New York, 1994.

Griffin, M.D., and French, J.R., *Space Vehicle Design*, American Institute of Aeronautics and Astronautics Education Series, New York, 1991.

Hale, Francis J., *Introduction to Space Flight*, Prentice-Hall, Inc., Englewood Cliffs, 1994.

Kourtides, D.A., and Pitts, W.C., "Composite Multilayer Insulations for Thermal Protection of Aerospace Vehicles," National Aeronautics and Space Administration TM-101071, February 1989.

Larson, Wiley J., and Wertz J.R., *Space Mission Analysis and Design*, Second Edition, Kluwer Academic Publications and Microcosm, Inc., Torrance, 1992.

Lyne, J.E., "Effect of Parking Orbit Period on Aerocapture for Manned Mars Missions," American Institute of Aeronautics and Astronautics Paper 92-103, February 1992.

Lyne, J.E., "Physiologically Constrained Aerocapture for Manned Mars Missions," National Aeronautics and Space Administration TM-103954, August 1992.

Lyne, J.E., Anagnost, A., and Tauber, M.E., "Parametric Study of Manned Aerocapture Part II: Mars Entry," American Institute of Aeronautics and Astronautics Paper 91-2871, August 1991.

Lyne, J.E., Tauber, M.E., and Braun, R.D., "Parametric Study of Manned Aerocapture Part I: Earth Return from Mars," *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, New Orleans, Louisiana, August 1991.

Lyons, Daniel T., "Aerobraking Magellan: Plan versus Reality," American Astrodynamics Specialist Paper 94-118, 1994.

Mars Observer Planetary Constants and Models, Jet Propulsion Lab D-3444, November 1990.

Mars Surveyor Program 2001 Orbiter Mission, Proposal Information Package, Jet Propulsion Laboratory, Pasadena, June 1997.

Mease, K.D., "Optimization of Aeroassisted Orbital Transfer: Current Status," Volume 36, Numbers 1/2, January-June 1988, pp. 7-33.

Proceedings of NASA Jet Propulsion Laboratory's Workshop on Aeroassisted Technologies, Pasadena, California, 1997.

Prussing, J.E., and Conway, B.A., *Orbital Mechanics*, Oxford University Press, New York, 1993.

Regan, F.J., *Re-Entry Vehicle Dynamics*, American Institute of Aeronautics and Astronautics Education Series, New York, 1984.

Ross, I. Michael, *3-DOF Model: Canonical EOMs Notes*, Monterey, February 1998.

Spriesterbach, T.P., *Performance Analysis of Non-Coplanar Synergetic Maneuvers*, Master's Thesis, Naval Postgraduate School, 1991.

Tauber, M.E., Bowles, J.V., and Yang, L., "The Use of Atmospheric Braking During Mars Missions," Journal of Spacecraft and Rockets, Volume 27, Number 5, 1990, pp. 514-521.

Tauber, M.E., and Seiff, A., "Optimizations of Heating of Conical Bodies Making Lifting Hyperbolic Entries into the Atmospheres of Earth and Mars," *Proceedings of the AIAA Entry Technology Conference*, Williamsburg, Virginia, October 1964.

Tauber, M.E., and Yang, L., "Performance Comparison of Maneuvering Vehicles Returning from Orbit," American Institute of Aeronautics and Astronautics Paper 87-2490, August 1987.

Using MATLAB, The MathWorks, Inc., Natick, 1997, pp 8-2 to 8-46.

Vallado, David A., *Fundamentals of Astrodynamics and Applications*, Mcgraw-Hill Companies, Inc., New York, 1997, pp. 490-510.

Vinh, N.X., Busemann, A., and Culp, R., *Hypersonic and Planetary Flight Mechanics*, Univ. of Michigan Press, Ann Arbor, Michigan, 1980, pp. 100-107.

Walberg, Gerald D., "A Survey of Aeroassisted Orbit Transfer," *Proceedings of the AIAA 9th Atmospheric Flight Mechanics Conference*, San Diego, California, August 1982.

Wercinski, P.F., and Lyne, J.E., "Mars Aerocapture: Extension and Refinement," *Journal of Spacecraft and Rockets*, Volume 31, Number 4, 1993, pp. 703-705.

Willcockson, W.H., "Magellan Aerobraking Control Corridor Design and Implementation," American Astrodynamics Specialist Paper 94-117, 1994.

Zike, Steve, *Mission Analysis for the Mars 2007 Opportunity*, Master's Thesis, Naval Postgraduate School, Monterey, California, December 1998.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center.....2
   8725 John J. Kingman Rd. STE 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library.....2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, CA 93943-5101

3. Department Chairman, Code AA..... 1
   Department of Aeronautics and Astronautics
   Naval Postgraduate School
   Monterey, CA 93943-5000

4. Department of Aeronautics and Astronautics..... 2
   ATTN: Professor I. Michael Ross, Code AA/Ro
   Naval Postgraduate School
   Monterey, CA 93943-5000

5. Mr. Steven E. Matousek.....5
   NASA Jet Propulsion Laboratory, Code 301-160
   4800 Oak Grove Dr.
   Pasadena, CA 91109-8099

6. LCDR (Sel.) Zigmond V. Leszczynski, USN..... 3
   640 Cove Terrace
   Arnold, MD 21012